5. Logik (Teil II) – Logik erster Stufe (Prädikatenlogik)

In Kapitel 3 haben wir bereits die Aussagenlogik kennengelernt, die einen Formalismus darstellt, mit dessen Hilfe man "Wissen" modellieren und Schlüsse aus dem Wissen ziehen kann.

In diesem Kapitel werden wir die Logik erster Stufe (bzw. Prädikatenlogik) als einen weiteren solchen Formalismus kennenlernen. Im Vergleich zur Aussagenlogik hat die Prädikatenlogik den Vorteil, dass

- eine klare Trennung zwischen "Daten" einerseits und "Logik" andererseits besteht, und dass in der Prädikatenlogik
- wesentlich umfangreichere Ausdrucksmöglichkeiten zur Verfügung stehen.

Der Preis für diese Vorteile ist allerdings, dass die Prädikatenlogik algorithmisch deutlich schwerer zu handhaben ist als die Aussagenlogik.

5.1 Motivation zur Logik erster Stufe

Grenzen der Aussagenlogik

Beispiel 5.1 (Verwandtschaftsbeziehungen)

Die Aussagenlogik kann helfen, um Aussagen der Art

"Anne und Bernd sind Geschwister.
Wenn Christine Annes Tochter ist, dann ist Bernd Christines Onkel"

zu modellieren und Schlüsse daraus zu ziehen.

Für die Modellierung der folgenden Aussage ist die Aussagenlogik aber eher ungeeignet:

"Es gibt in Frankfurt mindestens 2 Leute, die mehr als 3 Kinder, aber selbst keine Geschwister haben."

Beispiel 5.2 (Arithmetische Aussagen)

Die Aussagenlogik kann helfen, um Sätze der Art

"Wenn eine Zahl gerade ist, dann ist sie nicht ungerade"

zu formalisieren.

Für viele andere Aussagen ist die Aussagenlogik aber eher ungeeignet, zum Beispiel:

"Es gibt eine Zahl, die nicht Summe zweier Primzahlen ist."

Ein Überblick über die Logik erster Stufe

Die Logik erster Stufe ist ein Formalismus, mit dem man die in den beiden obigen Beispielen genannten Aussagen bequem beschreiben kann.

Genau wie die Aussagenlogik besitzt die Logik erster Stufe

- eine Syntax, die festlegt, welche Zeichenketten Formeln der Logik erster Stufe sind und

- eine <u>Semantik</u>, die festlegt, welche "Bedeutung" einzelne Formeln haben.

Die Logik erster Stufe beschäftigt sich mit <u>Objekten</u> (z.B. den Einwohnern Frankfurts und deren Verwandtschaftsbeziehungen (Bsp 5.1) oder den natürlichen Zahlen und deren Addition und Multiplikation (Bsp 5.2)) und <u>Aussagen über deren Eigenschaften</u>.

(Im Gegensatz dazu beschäftigt sich die Aussagen-logik nicht mit Objekten sondern lediglich mit "wahren" und "falschen" Aussagen und deren Kombination.)

Vor der Einführung in die Syntax und die Semantik der Logik erster Stufe, wenden wir uns zunächst den Objekten zu, über die Formeln der Logik erster Stufe "reden" können.

5.2 Strukturen

Die Objekte, über die Formeln der Logik erster Stufe Aussagen treffen können, heißen <u>Strukturen</u>.

Viele Objekte lassen sich auf natürliche Weise durch solche Strukturen repräsentieren, beispielsweise

- Graphen $G = (V, E)$
- Bäume $B = (V, E)$
- die natürlichen Zahlen mit Addition und Multiplikation, $(\mathbb{N}, +, \times)$
- die reellen Zahlen mit Addition, Multiplikation und den Konstanten 0 und 1, $(\mathbb{R}, +, \times, 0, 1)$
- Datenbanken

usw.

Die im Folgenden definierten <u>Signaturen</u> legen den "Typ" (bzw. das "Format") der entsprechenden Strukturen fest.

Definition 5.3

Eine <u>Signatur</u> (bzw. ein <u>Vokabular</u> bzw. eine <u>Symbolmenge</u>; englisch: signature, vocabulary) ist eine Menge σ (in Worten: sigma) von Relationssymbolen, Funktionssymbolen und/oder Konstantensymbolen

Jedes Relationssymbol $\dot{R} \in \sigma$ und jedes Funktionssymbol $\dot{f} \in \sigma$ hat eine <u>Stelligkeit</u> (bzw Arität, engl.: arity)

$$ar(\dot{R}) \in \mathbb{N}_{>0} \quad \text{bzw} \quad ar(\dot{f}) \in \mathbb{N}_{>0}.$$

Notation 5.4

- In diesem Kapitel bezeichnet der griechische Buchstabe σ immer eine Signatur.
- Wir kennzeichnen Symbole aus σ immer mit einem Punkt, wie in $\dot{R}$ bzw. $\dot{f}$.
- Für Relationssymbole verwenden wir meistens Großbuchstaben wie $\dot{R}, \dot{P}, \dot{E}, \dot{R}_1, \dot{R}_2, \dots$,
  für Funktionssymbole verwenden wir meistens Kleinbuchstaben wie $\dot{f}, \dot{g}, \dot{h}, \dots$,

- Für Konstantensymbole verwenden wir meistens Kleinbuchstaben wie $\dot{c}, \dot{d}, \ldots$.

- Gelegentlich verwenden wir als Relations- und Funktionssymbole auch Zeichen wie

  $\dot{\leq}$ (2-stelliges Relationssymbol),

  $\dot{+}, \dot{\times}$ (2-stellige Funktionssymbole),

  $\dot{0}, \dot{1}$ (Konstantensymbole).

Definition 5.5

Eine <u>Struktur über der Signatur σ</u> (kurz: <u>σ-Struktur</u>) ist ein Paar

$$\mathfrak{A} = (A, \alpha),$$

bestehend aus:

- einer nicht-leeren Menge A, dem so genannten <u>Universum</u> (bzw. <u>Träger</u>, engl.: universe, domain) von $\mathfrak{A}$, und

- einer auf σ definierten Abbildung α, die
  - jedem Relationssymbol $\dot{R} \in \sigma$ eine Relation $\alpha(\dot{R}) \subseteq A^{ar(\dot{R})}$ der Stelligkeit $ar(\dot{R})$ zuordnet

- jedem Funktionssymbol $\dot{f} \in \sigma$ eine Funktion $\alpha(\dot{f}) : A^{ar(\dot{f})} \to A$ zuordnet,

- jedem Konstantensymbol $\dot{c} \in \sigma$ ein Element $\alpha(\dot{c}) \in A$ zuordnet.

Notation 5.6

- Strukturen bezeichnen wir meistens mit Fraktur-Buchstaben $\mathfrak{A}, \mathfrak{B}, \mathfrak{G}, \ldots$; das Universum der Strukturen durch die entsprechenden lateinischen Großbuchstaben, also $A, B, G, \ldots$

- Ist $\mathfrak{A} = (A, \alpha)$ eine σ-Struktur, so schreiben wir für jedes Symbol $\dot{S} \in \sigma$ oft

$$\dot{S}^{\mathfrak{A}} \quad \text{an Stelle von} \quad \alpha(\dot{S}).$$

An Stelle von $\mathfrak{A} = (A, \alpha)$ schreiben wir oft auch $\mathfrak{A} = (A, (\dot{S}^{\mathfrak{A}})_{\dot{S} \in \sigma})$,

oder, falls $\sigma = \{\dot{R}_1, \ldots, \dot{R}_k, \dot{f}_1, \ldots, \dot{f}_\ell, \dot{c}_1, \ldots, \dot{c}_m\}$ ist,

$$\mathfrak{A} = (A, \dot{R}_1^{\mathfrak{A}}, \ldots, \dot{R}_k^{\mathfrak{A}}, \dot{f}_1^{\mathfrak{A}}, \ldots, \dot{f}_\ell^{\mathfrak{A}}, \dot{c}_1^{\mathfrak{A}}, \ldots, \dot{c}_m^{\mathfrak{A}}).$$

Beispiel 5.7 (Arithmetische Strukturen)

Sei $\sigma_{Ar} := \{ \dot{+}, \dot{\times}, \dot{0}, \dot{1} \}$, wobei

$\dot{+}$ und $\dot{\times}$ 2-stellige Funktionssymbole und
$\dot{0}$ und $\dot{1}$ Konstantensymbole sind.

Wir betrachten die σ_{Ar}-Struktur

$$\mathcal{N} := \left(\mathbb{N}, \dot{+}^{\mathcal{N}}, \dot{\times}^{\mathcal{N}}, \dot{0}^{\mathcal{N}}, \dot{1}^{\mathcal{N}} \right),$$

wobei $\dot{+}^{\mathcal{N}}$ und $\dot{\times}^{\mathcal{N}}$ die natürliche Addition bzw. Multiplikation auf $\mathbb{N}$ sind und $\dot{0}^{\mathcal{N}} := 0$, $\dot{1}^{\mathcal{N}} := 1$.

Entsprechend können wir σ_{Ar}-Strukturen $\mathcal{Z}, \mathcal{Q}, \mathcal{R}$ mit Universum $\mathbb{Z}, \mathbb{Q}, \mathbb{R}$ definieren.

Beispiel 5.8 (Graphen und Bäume)

Sei $\sigma_{Graph} := \{ \dot{E} \}$, wobei $\dot{E}$ ein 2-stelliges Relationssymbol ist.

Jeder gerichtete Graph bzw gerichtete Baum (V, E) lässt sich als σ_{Graph}-Struktur $\mathfrak{A} = (A, E^{\mathfrak{A}})$ mit Universum $A := V$ und Relation $\dot{E}^{\mathfrak{A}} := E$ auffassen

Beispiel 5.9 (Ordnungen)

Sei $\sigma_{Ord} := \{ \dot{\leq} \}$, wobei $\dot{\leq}$ ein 2-stelliges Relationssymbol ist.

Jeder Präordnung, partiellen Ordnung oder linearen Ordnung $\leq$ auf einer Menge A entspricht eine σ_{Ord}-Struktur

$$\mathfrak{A} = (A, \dot{\leq}^{\mathfrak{A}})$$

mit $\dot{\leq}^{\mathfrak{A}} := \leq$.

Frage: Wann sind zwei σ-Strukturen $\mathfrak{A}$ und $\mathfrak{B}$ "prinzipiell gleich" (Fachbegriff: isomorph)?

Antwort: Falls $\mathfrak{B}$ aus $\mathfrak{A}$ entsteht, indem man die Elemente des Universums von $\mathfrak{A}$ umbenennt.

Analog zum Begriff der Isomorphie von Graphen (Def. 4.26) wird dies durch folgende Definition präzisiert:

<u>Definition 5.10</u>

Sei σ eine Signatur und seien $\mathfrak{A}$ und $\mathfrak{B}$ zwei σ-Strukturen.

$\mathfrak{A}$ und $\mathfrak{B}$ heißen <u>isomorph</u> (kurz: $\mathfrak{A} \cong \mathfrak{B}$, in Worten: $\mathfrak{A}$ ist isomorph zu $\mathfrak{B}$), falls es eine <u>bijektive</u> Abbildung $\pi : A \to B$ gibt, für die gilt:

- für jedes Relationssymbol $\dot{R} \in \sigma$, für $r := ar(\dot{R})$ und für alle r-Tupel $(a_1, \dots, a_r) \in A^r$ gilt:
$$(a_1, \dots, a_r) \in R^{\mathfrak{A}} \iff \big(\pi(a_1), \dots, \pi(a_r)\big) \in R^{\mathfrak{B}}$$
- für jedes Konstantensymbol $\dot{c} \in \sigma$ gilt
$$\pi(c^{\mathfrak{A}}) = c^{\mathfrak{B}}$$
- für jedes Funktionssymbol $\dot{f} \in \sigma$, für $r := ar(\dot{f})$ und für alle r-Tupel $(a_1, \dots, a_r) \in A^r$ gilt:
$$\pi\big(f^{\mathfrak{A}}(a_1, \dots, a_r)\big) = f^{\mathfrak{B}}\big(\pi(a_1), \dots, \pi(a_r)\big).$$

Eine solche Abbildung π wird <u>Isomorphismus von $\mathfrak{A}$ nach $\mathfrak{B}$</u> genannt.

Beispiel 5.11

(a) Ist $A = \{1,2,3,4\}$, $B = \{6,7,8,9\}$, und sind $\dot{\leq}^{\mathfrak{A}}$ und $\dot{\leq}^{\mathfrak{B}}$ die natürlichen linearen Ordnungen auf A und B, so sind die beiden σ_{Ord}-Strukturen $\mathfrak{A} = (A, \dot{\leq}^{\mathfrak{A}})$ und $\mathfrak{B} = (B, \dot{\leq}^{\mathfrak{B}})$ isomorph.

Skizze:

Allgemein gilt: Sind A und B endliche Mengen mit $|A| = |B|$ und sind $\dot{\leq}^{\mathfrak{A}}$ und $\dot{\leq}^{\mathfrak{B}}$ lineare Ordnungen auf $\mathfrak{A} = (A, \dot{\leq}^{\mathfrak{A}})$ und $\mathfrak{B} = (B, \dot{\leq}^{\mathfrak{B}})$, so ist $\mathfrak{A} \cong \mathfrak{B}$, und die Abbildung π, die das (bzgl $\dot{\leq}^{\mathfrak{A}}$) kleinste Element in A auf das (bzgl. $\dot{\leq}^{\mathfrak{B}}$) kleinste Element von $\mathfrak{B}$ abbildet und, allgemein, für jedes $i \in \{1, \ldots, |A|\}$ das i-kleinste Element in A (bzgl $\dot{\leq}^{\mathfrak{A}}$) auf das i-kleinste Element in B (bzgl. $\dot{\leq}^{\mathfrak{B}}$) abbildet, ein Isomorphismus von $\mathfrak{A}$ nach $\mathfrak{B}$.

(b) Sind $\dot{\leq}^{\mathbb{N}}$ und $\dot{\leq}^{\mathbb{Z}}$ die natürlichen linearen Ordnungen auf $\mathbb{N}$ und $\mathbb{Z}$, so sind die σ_{Ord}-Strukturen

$$\mathfrak{A} := (\mathbb{N}, \dot{\leq}^{\mathbb{N}}) \quad \text{und} \quad \mathfrak{B} := (\mathbb{Z}, \dot{\leq}^{\mathbb{Z}})$$

nicht isomorph (kurz: $\mathfrak{A} \not\cong \mathfrak{B}$)

Skizze:

(c) Sei $\sigma := \{\dot{f}, \dot{c}\}$, wobei $\dot{f}$ ein 2-stelliges Funktionssymbol und $\dot{c}$ ein Konstantensymbol ist

Sei $\mathfrak{A} := (A, \dot{f}^{\mathfrak{A}}, \dot{c}^{\mathfrak{A}})$, wobei

- $A := \mathbb{N}$
- $\dot{f}^{\mathfrak{A}} := \dot{+}^{\mathbb{N}}$ die Addition auf $\mathbb{N}$
- $\dot{c}^{\mathfrak{A}} := \dot{0}^{\mathbb{N}}$ die natürliche Zahl 0 ist

und sei $\mathfrak{B} := (B, \dot{f}^{\mathfrak{B}}, \dot{c}^{\mathfrak{B}})$, wobei

- $B := \{2^n : n \in \mathbb{N}\}$ die Menge aller Zweierpotenzen
- $\dot{f}^{\mathfrak{B}} : B \times B \to B$ die Funktion mit
  $f^{\mathfrak{B}}(b_1, b_2) := b_1 \cdot b_2$, f.a. $b_1, b_2 \in B$
- $\dot{c}^{\mathfrak{B}} := 1 = 2^0 \in B$.

Dann gilt: $\mathfrak{A} \cong \mathfrak{B}$, und die Abbildung

$\pi : A \to B$ mit $\pi(n) := 2^n$, f.a. $n \in \mathbb{N}$,

ist ein Isomorphismus von $\mathfrak{A}$ nach $\mathfrak{B}$, denn:

- π ist eine bijektive Abbildung von A nach B
- Für das Konstantensymbol $\dot{c} \in \sigma$ gilt:

  $\pi(\dot{c}^{\mathfrak{A}}) \underset{\text{Def } \dot{c}^{\mathfrak{A}}}{=} \pi(0) \underset{\text{Def } \pi}{=} 2^0 \underset{\text{Def } \dot{c}^{\mathfrak{B}}}{=} \dot{c}^{\mathfrak{B}}$
- Für das Funktionssymbol $\dot{f} \in \sigma$ und für alle $(a_1, a_2) \in A^2$ gilt:

  $$\pi\Big(\dot{f}^{\mathfrak{A}}(a_1, a_2)\Big) \underset{\text{Def } \dot{f}^{\mathfrak{A}}}{=} \pi\big(a_1 + a_2\big) \underset{\text{Def } \pi}{=} 2^{a_1 + a_2}$$

  und

  $$\dot{f}^{\mathfrak{B}}\big(\pi(a_1), \pi(a_2)\big) \underset{\text{Def } \pi}{=} \dot{f}^{\mathfrak{B}}\big(2^{a_1}, 2^{a_2}\big) \underset{\text{Def } \dot{f}^{\mathfrak{B}}}{=} 2^{a_1} \cdot 2^{a_2} = 2^{a_1 + a_2}.$$

  Also: $\pi\big(\dot{f}^{\mathfrak{A}}(a_1, a_2)\big) = \dot{f}^{\mathfrak{B}}\big(\pi(a_1), \pi(a_2)\big)$.

Somit ist π ein Isomorphismus von $\mathfrak{A}$ nach $\mathfrak{B}$.

Wir wissen nun, über welche Objekte Formeln der Logik erster Stufe "reden" können: über σ-Strukturen, wobei σ eine Signatur ist.

Als nächstes legen wir die Syntax der Logik erster Stufe fest.

5.3 Syntax der Logik erster Stufe

Die Logik erster Stufe übernimmt, verändert und erweitert die Syntax der Aussagenlogik

- Was gleich bleibt:
  Alle Junktoren $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ werden übernommen
- Was sich ändert:
  - Variablen stehen nicht mehr für "wahre" oder "falsche" Aussagen, sondern für Elemente im Universum einer σ-Struktur
  - Variablen sind keine atomaren Formeln mehr
- Was neu hinzukommt
  - Es gibt <u>Quantoren</u> $\exists$ (für "es existiert") und $\forall$ (für "für alle")
  - Es gibt Symbole für Elemente aus der Signatur σ.

Definition 5.12 (Variablen und Alphabet der Logik erster Stufe)

(a) Eine **Individuenvariable** (kurz: **Variable**) hat die Form v_i, für $i \in \mathbb{N}$.

Die Menge aller Variablen bezeichnen wir mit Var, d.h. $\text{Var} := \{ v_i : i \in \mathbb{N} \}$

(b) Sei σ eine Signatur.
Das **Alphabet A_σ** der **Logik erster Stufe über σ** besteht aus

- den Variablen in Var
- den Symbolen in σ
- den Quantoren $\exists$ (Existenzquantor) und $\forall$ (Allquantor)
- dem Gleichheitssymbol $\doteq$
- den Junktoren $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- den Klammern (,) und dem Komma ,

D.h.

$$A_\sigma = \text{Var} \cup \sigma \cup \{\exists, \forall\} \cup \{\doteq\} \,\dot{\cup}\, \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow, (,), ,\}$$

Definition 5.13 (Terme der Logik erster Stufe)

Sei σ eine Signatur.
Die Menge T_σ der σ-Terme ist die folgendermaßen rekursiv definierte Teilmenge von A_σ^*:

Basisregeln:
- Für jedes Konstantensymbol $\dot{c} \in \sigma$ ist $\dot{c} \in T_\sigma$
- Für jede Variable $x \in Var$ ist $x \in T_\sigma$

Rekursive Regeln:
- Für jedes Funktionssymbol $\dot{f} \in \sigma$ und für $r := ar(\dot{f})$ gilt:
  Sind $t_1 \in T_\sigma, \ldots, t_r \in T_\sigma$, so ist auch $\dot{f}(t_1, \ldots, t_r) \in T_\sigma$.

Beispiel 5.14

Sei $\sigma = \{\dot{f}, \dot{c}\}$ die Signatur aus Bsp. 5.11 (c), die aus einem 2-stelligen Funktionssymbol $\dot{f}$ und einem Konstantensymbol $\dot{c}$ besteht.

Folgende Worte aus A_σ^* sind σ-Terme:
$\dot{c}$, v_4, $\dot{f}(\dot{c}, \dot{c})$, $\dot{f}(\dot{c}, v_0)$, $\dot{f}(\dot{c}, \dot{f}(\dot{c}, v_0))$

Folgende Worte sind keine σ-Terme:
0, $\dot{f}(0, \dot{c})$, $\dot{f}(v_0, \dot{c}, v_1)$, $f^{\mathfrak{A}}(2,3)$.

Definition 5.15 (Formeln der Logik erster Stufe)

Sei σ eine Signatur.

Die Menge $\mathrm{FO}[\sigma]$ aller Formeln der Logik erster Stufe über der Signatur σ (kurz: $\mathrm{FO}[\sigma]$-Formeln; FO steht für die englische Bezeichnung der Logik erster Stufe: first-order logic) ist die folgendermaßen rekursiv definierte Teilmenge von A_σ^*:

Basisregeln:

- Für alle σ-Terme t_1 und t_2 in T_σ gilt:
  $t_1 \doteq t_2 \in \mathrm{FO}[\sigma]$
- Für jedes Relationssymbol $\dot{R} \in \sigma$, für $r := ar(\dot{R})$ und für alle σ-Terme $t_1, \dots, t_r$ in T_σ gilt:
  $\dot{R}(t_1, \dots, t_r) \in \mathrm{FO}[\sigma]$.

Bemerkung: $\mathrm{FO}[\sigma]$-Formeln der Form $t_1 \doteq t_2$ oder $\dot{R}(t_1, \dots, t_r)$ heißen atomare σ-Formeln.

Rekursive Regeln:

- Ist $\varphi \in \mathrm{FO}[\sigma]$, so auch $\neg\varphi \in \mathrm{FO}[\sigma]$
- Ist $\varphi \in \mathrm{FO}[\sigma]$ und $\psi \in \mathrm{FO}[\sigma]$, so ist auch
  - $(\varphi \wedge \psi) \in \mathrm{FO}[\sigma]$
  - $(\varphi \vee \psi) \in \mathrm{FO}[\sigma]$

- $(\varphi \to \psi) \in \mathrm{FO}[\sigma]$
- $(\varphi \leftrightarrow \psi) \in \mathrm{FO}[\sigma]$

- Ist $\varphi \in \mathrm{FO}[\sigma]$ und ist $x \in \mathrm{Var}$, so ist auch
  - $\exists x \varphi \in \mathrm{FO}[\sigma]$
  - $\forall x \varphi \in \mathrm{FO}[\sigma]$.

Beispiel 5.16

(a) Sei $\sigma = \{\dot{f}, \dot{c}\}$ die Signatur aus Bsp. 5.14, die aus einem 2-stelligen Funktionssymbol $\dot{f}$ und einem Konstantensymbol $\dot{c}$ besteht

Folgende Worte aus A_σ^* sind $\mathrm{FO}[\sigma]$-Formeln:

- $\dot{f}(v_0, v_1) \doteq \dot{c}$ (atomare σ-Formel)
- $\forall v_2\ \dot{f}(v_2, \dot{c}) \doteq v_2$
- $\neg \exists v_3 \left(\dot{f}(v_3, v_3) \doteq v_3 \wedge \neg v_3 \doteq \dot{c}\right)$

Folgende Worte sind keine $\mathrm{FO}[\sigma]$-Formeln:

- $(\dot{f}(v_0, v_1) \doteq \dot{c})$
- $(\forall v_2 (\dot{f}(v_2, \dot{c}) \doteq v_2))$
- $\exists \dot{c}\ \dot{f}(v_0, \dot{c}) \doteq v_0$

(b) Sei $\sigma_{Graph} = \{\dot{E}\}$ die Signatur, die aus einem 2-stelligen Relationssymbol besteht. Folgendes ist eine $\mathrm{FO}[\sigma_{Graph}]$-Formel:

$$\forall v_0 \forall v_1 \left(\left(\dot{E}(v_0, v_1) \wedge \dot{E}(v_1, v_0) \right) \to v_0 \doteq v_1 \right)$$

Intuition zur Semantik:

In einem Graphen $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ sagt diese Formel folgendes aus:

"für alle Knoten $a_0 \in A$ und
für alle Knoten $a_1 \in A$ gilt:
falls $(a_0, a_1) \in \dot{E}^{\mathfrak{A}}$ und $(a_1, a_0) \in \dot{E}^{\mathfrak{A}}$,
so ist $a_0 = a_1$".

Die Formel sagt in einem Graph $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ also gerade aus, dass die Kantenrelation $\dot{E}^{\mathfrak{A}}$ antisymmetrisch ist (vgl. Definition 4.65).

D.h.: Ein Graph $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ <u>erfüllt</u> die Formel genau dann, wenn die Kantenrelation $\dot{E}^{\mathfrak{A}}$ antisymmetrisch ist.

Notation 5.17

- Statt mit $v_0, v_1, v_2, \ldots$ bezeichnen wir Variablen oft auch mit $x, y, z, \ldots$ oder mit Varianten wie $x', y_1, y_2, \ldots$.

- Für gewisse 2-stellige Funktionssymbole wie $\dot{+}, \dot{\times} \in \sigma_{Ar}$ oder $\dot{\leq} \in \sigma_{Ord}$ verwenden wir Infix- statt Präfixschreibweise und setze Klammern dabei auf natürliche Weise, um die eindeutige Lesbarkeit zu gewährleisten.

  Beispiel: An Stelle des (formal korrekten) Terms

  $\dot{\times}\big(\dot{+}(x,y), z\big)$ schreiben wir $(x \dot{+} y) \dot{\times} z$.

  An Stelle der (formal korrekten) atomaren Formel

  $\dot{\leq}(x,y)$ schreiben wir $x \dot{\leq} y$.

Wir wissen nun, welche Zeichenketten (über dem Alphabet A_σ) FO[σ]-Formeln genannt werden (Syntax). Bevor wir die formale Definition der Semantik angeben, betrachten wir zunächst einige Beispiele, um ein "intuitives Verständnis" der Semantik zu bekommen.

5.4 Beispiele zur Semantik der Logik erster Stufe

Beispiel 5.18 (gerichtete Graphen)

Sei $\sigma_{Graph} = \{ \dot{E} \}$, wobei $\dot{E}$ ein 2-stelliges Relationssymbol ist.

(a) Die $FO[\sigma_{Graph}]$ - Formel

$$\varphi := \forall x \, \forall y \, \big(\dot{E}(x,y) \rightarrow \dot{E}(y,x) \big)$$

besagt:

"Für alle Knoten x und für alle Knoten y gilt: Falls es eine Kante von x nach y gibt, so gibt es auch eine Kante von y nach x"

Für jeden Graphen $\mathcal{A} = (A, \dot{E}^{\mathcal{A}})$ gilt daher:

$$\mathcal{A} \text{ erfüllt } \varphi \iff \dot{E}^{\mathcal{A}} \text{ ist symmetrisch}$$

Umgangssprachlich sagen wir auch:

"Die Formel φ sagt in einem Graphen $\mathcal{A}$ aus, dass dessen Kantenrelation symmetrisch ist"

(b) Die folgende $FO[\sigma_{Graph}]$-Formel drückt aus, dass es von Knoten x zu Knoten y einen Weg der Länge 3 gibt:

$$\varphi(x,y) := \exists z_1 \exists z_2 \Big(\big(E(x,z_1) \wedge E(z_1,z_2) \big) \wedge E(z_2,y) \Big)$$

(c) Der $FO[\sigma]$-Satz

$$\forall x \forall y \exists z_1 \exists z_2 \Big(\big(E(x,z_1) \wedge E(z_1,z_2) \big) \wedge E(z_2,y) \Big)$$

sagt in einem Graphen $\mathfrak{A}$ aus, dass es zwischen je 2 Knoten einen Weg der Länge 3 gibt.

Beispiel 5.19 (Verwandtschaftsbeziehungen)

Um Verwandtschaftsbeziehungen zu modellieren, können wir die Symbolmenge σ benutzen, die aus den folgenden Symbolen besteht:

- 1-stellige Funktionen Vater, Mutter

  (Bedeutung: $x = \text{Vater}(y)$ besagt "x ist der Vater von y")
- 2-stellige Relationen Geschwister, Vorfahr

  (Bedeutung: Geschwister(x,y) besagt, dass x und y Geschwister sind,

  Vorfahr(x,y) besagt, dass x ein Vorfahr von y ist)

Generelles Wissen über Verwandtschaftsbeziehungen lässt sich durch Formeln der Logik erster Stufe repräsentieren, beispielsweise:

- "Personen mit gleichem Vater und gleicher Mutter sind Geschwister":

$$\forall x \forall y \Big(\big(\mathrm{Vater}(x) = \mathrm{Vater}(y) \land \mathrm{Mutter}(x) = \mathrm{Mutter}(y) \big) \rightarrow \mathrm{Geschwister}(x,y) \Big)$$

- "Eltern sind gerade die unmittelbaren Vorfahren":

$$\forall x \forall y \Big(\big(x = \mathrm{Vater}(y) \lor x = \mathrm{Mutter}(y) \big) \leftrightarrow \big(\mathrm{Vorfahr}(x,y) \land \neg \exists z \, (\mathrm{Vorfahr}(x,z) \land \mathrm{Vorfahr}(z,y)) \big) \Big)$$

- "Die Relation Vorfahr ist transitiv"

$$\forall x \forall y \forall z \Big(\big(\mathrm{Vorfahr}(x,y) \land \mathrm{Vorfahr}(y,z) \big) \rightarrow \mathrm{Vorfahr}(x,z) \Big)$$

- Die folgende Formel $\varphi(x,y)$ besagt, dass x Tante oder Onkel von y ist:

$$\varphi(x,y) := \exists z \Big(\mathrm{Geschwister}(x,z) \land \big(z \doteq \mathrm{Vater}(y) \lor z \doteq \mathrm{Mutter}(y) \big) \Big)$$

- Die folgende Formel $\psi(x)$ besagt, dass x Vater von genau 2 Kindern ist:

$$\psi(x) := \exists y_1 \exists y_2 \Big(\big((x = \mathrm{Vater}(y_1) \land x \doteq \mathrm{Vater}(y_2)) \land \neg\, y_1 \doteq y_2 \big) \land \forall z \, \big(x = \mathrm{Vater}(z) \rightarrow (z \doteq y_1 \lor z \doteq y_2) \big) \Big).$$

5.5 Semantik der Logik erster Stufe

Um die formale Definition der Semantik der Logik erster Stufe angeben zu können, benötigen wir noch folgende Begriffe:

Notation 5.20

- Eine Formel ψ ist Teilformel einer Formel φ, wenn ψ als Teil-Wort in φ vorkommt

  Beispiel: $\psi := \dot{f}(v_0, v_1) \doteq \dot{c}$ ist Teilformel der Formel $\exists v_0\, \dot{f}(v_0, v_1) \doteq \dot{c}$.

- Ist φ eine Formel und x eine Variable, so heißt jedes Vorkommen von x in einer Teilformel der Form $\exists x \psi$ oder $\forall x \psi$ gebunden.

  Jedes andere Vorkommen von x in φ heißt frei.

  Beispiel: $\varphi := \left(\dot{f}(v_0, \dot{c}) \doteq v_3 \wedge \exists v_0\, \dot{f}(v_0, v_1) \doteq \dot{c}\right)$

  freies Vorkommen von v_0 in φ

  freies Vorkommen von v_3 in φ

  gebundene Vorkommen von v_0 in φ

  freies Vorkommen von v_1 in φ.

- Die Menge frei(φ) aller freien Variablen eine FO[σ]-Formel φ besteht aus allen Variablen, die mindestens einmal frei in φ vorkommen

  Beispiel:

  $$\text{frei}\left(\dot{f}(v_0, \dot{c}) \doteq v_3\right) = \{v_0, v_3\}$$

  $$\text{frei}\left(\exists v_0\, \dot{f}(v_0, v_1) \doteq \dot{c}\right) = \{v_1\}$$

  $$\text{frei}\left(\left(\dot{f}(v_0, \dot{c}) \doteq v_3 \wedge \exists v_0\, \dot{f}(v_0, v_1) \doteq \dot{c}\right)\right) = \{v_0, v_3, v_1\}$$

- Eine $\mathrm{FO}[\sigma]$-Formel φ heißt <u>Satz</u> (genauer: $\mathrm{FO}[\sigma]$-Satz), falls sie keine freien Variablen besitzt, d.h. falls $\mathrm{frei}(\varphi) = \emptyset$.

<u>Definition 5.21</u>: (Belegung und Interpretation)

(a) Eine <u>Belegung</u> in einer σ-Struktur $\mathfrak{A} = (A, \alpha)$ ist eine partielle Funktion β von Var nach A

(d.h. β ordnet jeder Variablen $x \in \mathrm{Def}(\beta)$ ein Element $\beta(x)$ aus dem Universum von $\mathfrak{A}$ zu).

(b) Eine Belegung β ist eine <u>Belegung für eine $\mathrm{FO}[\sigma]$-Formel φ</u> (bzw. <u>passend zu φ</u>), wenn $\mathrm{frei}(\varphi) \subseteq \mathrm{Def}(\beta)$.

(c) Eine <u>σ-Interpretation</u> ist ein Paar

$$\mathcal{I} = (\mathfrak{A}, \beta)$$

bestehend aus einer σ-Struktur $\mathfrak{A}$ und einer Belegung β in $\mathfrak{A}$.

$\mathcal{I} = (\mathfrak{A}, \beta)$ ist eine <u>Interpretation für eine $\mathrm{FO}[\sigma]$-Formel φ</u> (bzw. passend zu φ), wenn β passend zu φ ist.

Definition 5.22 (Semantik von σ-Termen)

Sei σ eine Signatur.

Rekursiv über den Aufbau von T_σ definieren wir eine Funktion $[\![\cdot]\!]^{\cdot}$, die jedem σ-Term $t \in T_\sigma$ und jeder σ-Interpretation $\mathcal{I} = (\mathfrak{A}, \beta)$, so dass $Def(\beta)$ jede in t vorkommende Variable enthält, einen Wert $[\![t]\!]^{\mathcal{I}} \in A$ zuordnet:

- Für alle $x \in Var$ ist $[\![x]\!]^{\mathcal{I}} := \beta(x)$
- Für alle Konstantensymbole $\dot{c} \in \sigma$ ist $[\![\dot{c}]\!]^{\mathcal{I}} := \dot{c}^{\mathfrak{A}}$
- Für alle Funktionssymbole $\dot{f} \in \sigma$, für $r := ar(\dot{f})$ und für alle σ-Terme $t_1, \dots, t_r \in T_\sigma$ gilt:

$$[\![\dot{f}(t_1, \dots, t_r)]\!]^{\mathcal{I}} := \dot{f}^{\mathfrak{A}}\left([\![t_1]\!]^{\mathcal{I}}, \dots, [\![t_r]\!]^{\mathcal{I}}\right).$$

Beispiel 5.23

Sei $\sigma = \{\dot{f}, \dot{c}\}$ und sei $\mathfrak{A} := (A, \dot{f}^{\mathfrak{A}}, \dot{c}^{\mathfrak{A}})$ mit

- $A := \mathbb{N}$
- $\dot{f}^{\mathfrak{A}} := \dot{+}^{\mathbb{N}}$ die Addition auf $\mathbb{N}$
- $\dot{c}^{\mathfrak{A}} := \dot{0}^{\mathbb{N}}$ die natürliche Zahl 0

wie in Beispiel 5.11 (c)

Sei β eine Belegung mit $\beta(v_1) = 1$ und $\beta(v_2) = 7$. Und sei $\mathcal{I} := (\mathfrak{A}, \beta)$.

Sei t der Term $\dot{f}(v_2, \dot{f}(v_1, \dot{c}))$

Dann gilt:

$$
\begin{aligned}
[\![t]\!]^{\mathcal{I}} &= [\![\dot{f}(v_2, \dot{f}(v_1, \dot{c}))]\!]^{\mathcal{I}} \\
&= \dot{f}^{\mathfrak{A}}\left([\![v_2]\!]^{\mathcal{I}}, [\![\dot{f}(v_1, \dot{c})]\!]^{\mathcal{I}}\right) \\
&\underset{\substack{\dot{f}^{\mathfrak{A}} = \text{Addition} \\ \text{auf } \mathbb{N}}}{=} [\![v_2]\!]^{\mathcal{I}} + [\![\dot{f}(v_1, \dot{c})]\!]^{\mathcal{I}} \\
&= \beta(v_2) + \dot{f}^{\mathfrak{A}}\left([\![v_1]\!]^{\mathcal{I}}, [\![\dot{c}]\!]^{\mathcal{I}}\right) \\
&= \beta(v_2) + \left([\![v_1]\!]^{\mathcal{I}} + [\![\dot{c}]\!]^{\mathcal{I}}\right) \\
&= \beta(v_2) + \beta(v_1) + \dot{c}^{\mathfrak{A}} \\
&\underset{\text{Def. } \beta \text{ und } \dot{c}^{\mathfrak{A}}}{=} 7 + 1 + 0 \\
&= 8
\end{aligned}
$$

Notation 5.24

- Ist β eine Belegung in einer σ-Struktur $\mathfrak{A}$, ist $x \in \mathrm{Var}$ und ist $a \in A$, so sei

$$\beta\frac{a}{x}$$

die Belegung mit $\mathrm{Def}(\beta\frac{a}{x}) := \mathrm{Def}(\beta) \cup \{x\}$, die für alle $y \in \mathrm{Def}(\beta\frac{a}{x})$ definiert ist durch

$$\beta\frac{a}{x}(y) := \begin{cases} a & \text{falls } y = x, \\ \beta(y) & \text{sonst.} \end{cases}$$

- Ist $\mathcal{I} = (\mathfrak{A}, \beta)$ eine σ-Interpretation, ist $x \in \mathrm{Var}$ und ist $a \in A$, so sei

$$\mathcal{I}\frac{a}{x} := (\mathfrak{A}, \beta\frac{a}{x}).$$

Wir können nun (endlich) die formale Semantik der Logik erster Stufe festlegen.

Definition 5.25 (Semantik der Logik erster Stufe)

Sei σ eine Signatur.
Rekursiv über den Aufbau von $FO[\sigma]$ definieren wir eine Funktion $[\![\cdot]\!]^{\cdot}$, die jeder $FO[\sigma]$-Formel φ und jeder zu φ passenden Interpretation $\mathcal{I} = (\mathcal{A}, \beta)$ einen Wahrheitswert (kurz: Wert) $[\![\varphi]\!]^{\mathcal{I}} \in \{0,1\}$ zuordnet:

Rekursionsanfang:

- Für alle σ-Terme t_1 und t_2 in T_σ gilt:

$$[\![t_1 \doteq t_2]\!]^{\mathcal{I}} = \begin{cases} 1 & \text{falls } [\![t_1]\!]^{\mathcal{I}} = [\![t_2]\!]^{\mathcal{I}} \\ 0 & \text{sonst} \end{cases}$$

- Für jedes Relationssymbol $\dot{R} \in \sigma$, für $r := ar(\dot{R})$ und für alle σ-Terme $t_1, \dots, t_r$ in T_σ gilt:

$$[\![R(t_1, \dots, t_r)]\!]^{\mathcal{I}} := \begin{cases} 1 & \text{falls } \left([\![t_1]\!]^{\mathcal{I}}, \dots, [\![t_r]\!]^{\mathcal{I}}\right) \in R^{\mathcal{A}} \\ 0 & \text{sonst.} \end{cases}$$

Rekursionsschritt

- Die Semantik der Junktoren $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ ist wie in der Aussagenlogik definiert — beispielsweise ist für $\varphi \in FO[\sigma]$ und $\psi \in FO[\sigma]$

$$[\![(\varphi \wedge \psi)]\!]^{\mathcal{I}} := \begin{cases} 1 & \text{falls } [\![\varphi]\!]^{\mathcal{I}} = 1 \text{ und } [\![\psi]\!]^{\mathcal{I}} = 1 \\ 0 & \text{sonst} \end{cases}$$

- Ist $\varphi \in \mathrm{FO}[\sigma]$ und ist $x \in \mathrm{Var}$, so ist
  - $$[\![\exists x\, \varphi]\!]^{\mathcal{I}} := \begin{cases} 1 & \text{falls es mindestens ein } a \in A \text{ gibt,} \\ & \text{so dass } [\![\varphi]\!]^{\mathcal{I}\frac{a}{x}} = 1 \\ 0 & \text{sonst} \end{cases}$$
  - $$[\![\forall x\, \varphi]\!]^{\mathcal{I}} := \begin{cases} 1 & \text{falls für alle } a \in A \text{ gilt:} \\ & [\![\varphi]\!]^{\mathcal{I}\frac{a}{x}} = 1 \\ 0 & \text{sonst} \end{cases}$$

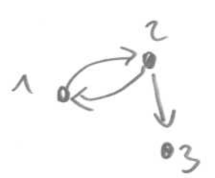
Beispiel 5.26

Sei $\sigma_{\mathrm{Graph}} = \{\dot{E}\}$ die Signatur, die aus einem 2-stelligen Relationssymbol besteht.

Betrachte die $\mathrm{FO}[\sigma]$-Formel

$$\varphi := \forall x \forall y \; \big(\dot{E}(x,y) \rightarrow \dot{E}(y,x)\big)$$

Sei $\mathfrak{A}$ die σ_{Graph}-Struktur, die den gerichteten Graphen



repräsentiert, d.h. $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ mit

$A = \{1, 2, 3\}$ und

$\dot{E}^{\mathfrak{A}} = \{(1,2), (2,1), (2,3)\}$.

Sei β die Belegung mit leerem Definitionsbereich, und sei $\mathcal{I} := (\mathfrak{A}, \beta)$.

Dann gilt:

$$[\![\varphi]\!]^{\mathcal{I}} = 1$$

$\Leftrightarrow$ für alle $a \in A$ gilt: $[\![\forall y\,(\dot{E}(x,y) \to \dot{E}(y,x))]\!]^{\mathcal{I}\frac{a}{x}} = 1$

$\Leftrightarrow$ für alle $a \in A$ gilt:
für alle $b \in B$ gilt: $[\![(\dot{E}(x,y) \to \dot{E}(y,x))]\!]^{\mathcal{I}\frac{a}{x}\frac{b}{y}} = 1$

$\Leftrightarrow$ für alle $a \in A$ und alle $b \in B$ gilt:
falls $[\![\dot{E}(x,y)]\!]^{\mathcal{I}\frac{a}{x}\frac{b}{y}} = 1$, so auch $[\![\dot{E}(y,x)]\!]^{\mathcal{I}\frac{a}{x}\frac{b}{y}} = 1$

$\Leftrightarrow$ für alle $a \in A$ und alle $b \in B$ gilt:
falls $(a,b) \in \dot{E}^{\mathfrak{A}}$, so auch $(b,a) \in \dot{E}^{\mathfrak{A}}$

$\Leftrightarrow$ $\dot{E}^{\mathfrak{A}}$ ist symmetrisch, vgl. Def. 4.65

Da in unserem konkreten Graphen $\mathfrak{A}$ für $a = 2$ und $b = 3$ gilt: $(a,b) \in \dot{E}^{\mathfrak{A}}$, aber $(b,a) \notin \dot{E}^{\mathfrak{A}}$, ist hier $[\![\varphi]\!]^{\mathcal{I}} = 0$.

Notation 5.27:

Sei σ eine Signatur und sei φ eine FO[σ]-Formel.

- Ist $\mathcal{I} = (\mathfrak{A}, \beta)$ eine zu φ passende σ-Interpretation, so sagen wir "$\mathcal{I}$ erfüllt φ" (bzw. "$\mathcal{I}$ ist ein Modell von φ", kurz: $\mathcal{I} \models \varphi$)

  falls $[\![\varphi]\!]^{\mathcal{I}} = 1$.

- Ist φ ein Satz (dh φ hat keine freien Variablen), so hängt die Tatsache, ob φ von einer Interpretation $\mathcal{I} = (\mathfrak{A}, \beta)$ erfüllt wird, nur von der Struktur $\mathfrak{A}$ und nicht von der Belegung β ab.

  An Stelle von "$\mathcal{I} \models \varphi$" schreiben wir dann kurz "$\mathfrak{A} \models \varphi$" und sagen "die σ-Struktur $\mathfrak{A}$ erfüllt den Satz φ".

Erfüllbarkeit, Allgemeingültigkeit, Folgerung und Äquivalenz

Definition 5.28:

Sei σ eine Signatur und sei φ eine $FO[\sigma]$-Formel.

(a) φ heißt <u>erfüllbar</u>, wenn es (mindestens) eine zu φ passende σ-Interpretation $\mathcal{I}$ gibt, die φ erfüllt.

(b) φ heißt <u>unerfüllbar</u>, wenn φ nicht erfüllbar ist.

(c) φ heißt <u>allgemeingültig</u>, wenn jede zu φ passende σ-Interpretation φ erfüllt.

Beispiel 5.29:

Sei $\sigma_{Graph} := \{\dot{E}\}$ die Signatur, die aus einem 2-stelligen Relationssymbol besteht.

Die $FO[\sigma]$-Formel $\varphi := \forall y\ \dot{E}(x,y)$ ist erfüllbar, aber nicht allgemeingültig, denn:

Sei $\mathcal{A} := (A, \dot{E}^{\mathcal{A}})$ der gerichtete Graph 

und sei β die Belegung mit $\beta(x) = 1$.

Dann erfüllt die Interpretation $(\mathfrak{A}, \beta)$ die Formel φ.
Somit ist φ erfüllbar.

Andererseits gilt für den Graphen $\mathfrak{B} := (B, \dot{E}^{\mathfrak{B}})$

und die Belegung β mit $\beta(x) = 1$, dass die zu φ passende σ-Interpretation $\mathcal{I} := (\mathfrak{B}, \beta)$ die Formel φ nicht erfüllt (d.h. $[\![\varphi]\!]^{\mathcal{I}} = 0$),

denn:

$[\![\varphi]\!]^{\mathcal{I}} = 1 \iff$ Für jedes $b \in B$ gilt: $[\![\dot{E}(x,y)]\!]^{\mathcal{I}\frac{b}{y}} = 1$

$\iff$ Für jedes $b \in B$ gilt:

$$\left(\beta\tfrac{b}{y}(x), \beta\tfrac{b}{y}(y)\right) \in \dot{E}^{\mathfrak{B}}$$

$\iff$ Für jedes $b \in B$ gilt:

$$(1, b) \in \dot{E}^{\mathfrak{B}}$$

Aber für $b := 1$ gilt: $(1,1) \notin \dot{E}^{\mathfrak{B}}$, und daher ist $[\![\varphi]\!]^{\mathcal{I}} = 0$. $\mathcal{I}$ ist also eine zu φ passende σ-Interpretation, die φ nicht erfüllt.
Somit ist φ nicht allgemeingültig.

Beobachtung 5.30:

Für alle Formeln φ der Logik erster Stufe gilt:

(a) φ ist allgemeingültig $\iff$ $\neg\varphi$ ist unerfüllbar.

(b) φ ist erfüllbar $\iff$ $\neg\varphi$ ist nicht allgemeingültig.

Beweis: Übung.

Definition 5.31 (semantische Folgerung)

Sei σ eine Signatur und seien φ und ψ zwei $FO[\sigma]$-Formeln.

Wir sagen: ψ folgt aus φ (kurz: $\varphi \models \psi$, "φ impliziert ψ"), falls für jede zu φ und ψ passende Interpretation $\mathcal{I}$ gilt:

$$\text{Falls } \underbrace{\mathcal{I} \models \varphi}_{\text{d.h. } [\![\varphi]\!]^{\mathcal{I}} = 1}, \text{ so auch } \underbrace{\mathcal{I} \models \psi}_{\text{d.h. } [\![\psi]\!]^{\mathcal{I}} = 1}.$$

Definition 5.32 (logische Äquivalenz)

Sei σ eine Signatur.
Zwei $FO[\sigma]$-Formeln φ und ψ heißen **äquivalent** (kurz: $\varphi \equiv \psi$), wenn für jede zu φ und ψ passende σ-Interpretation $\mathcal{I}$ gilt: $\mathcal{I}$ erfüllt φ $\iff$ $\mathcal{I}$ erfüllt ψ.

Beobachtung 5.33:

Sei σ eine Signatur und seien φ und ψ zwei $FO[\sigma]$-Formeln. Es gilt:

(a) $\varphi \equiv \psi \iff \varphi \models \psi$ und $\psi \models \varphi$

(b) $\varphi \equiv \psi \iff (\varphi \leftrightarrow \psi)$ ist allgemeingültig

(c) $\varphi \models \psi \iff (\varphi \to \psi)$ ist allgemeingültig.

Beweis: Übung.

Grenzen der Logik erster Stufe

In Beispiel 5.18 und 5.19 und auf den Übungsblättern 9 und 10 haben wir viele Beispiele für umgangssprachliche Aussagen kennengelernt, die man durch Formeln der Logik erster Stufe beschreiben kann. Es gibt allerdings auch Aussagen, die nicht in der Logik erster Stufe formalisiert werden können:

Satz 5.34:

Sei $\sigma_{Graph} := \{\dot{E}\}$ die Signatur, die aus einem 2-stelligen Relationssymbol $\dot{E}$ besteht. Es gilt:

(a) Es gibt keinen $FO[\sigma_{Graph}]$-Satz φ, so dass für jeden gerichteten Graphen $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ gilt:

$\mathfrak{A}$ erfüllt φ $\iff$ $\mathfrak{A}$ ist azyklisch (vgl. Definition 4.14)

(b) Es gibt keine $FO[\sigma_{Graph}]$-Formel ψ mit freien Variablen x und y, so dass für jeden gerichteten Graphen $\mathfrak{A} = (A, \dot{E}^{\mathfrak{A}})$ und jede zu ψ passende Belegung β in $\mathfrak{A}$ gilt:

$(\mathfrak{A}, \beta)$ erfüllt ψ $\iff$ es gibt in $\mathfrak{A}$ einen Weg von Knoten $\beta(x)$ zu Knoten $\beta(y)$.

Einen Beweis dieses Satzes können Sie in der Vorlesung "Logik in der Informatik" kennen lernen.

5.6 Ein Anwendungsbereich der Logik erster Stufe: Datenbanken

Relationale Datenbanken bestehen aus Tabellen, die sich als Relationen auffassen lassen. Datenbanken lassen sich daher als Strukturen über einer passenden Signatur auffassen.

Die in der Praxis gebräuchlichste Datenbank-anfragesprache ist SQL. Der "Kern" von SQL basiert auf der Logik erster Stufe, die in der Datenbankterminologie oft auch als "relationaler Kalkül" (engl.: "relational calculus") bezeichnet wird.

Beispieldatenbank mit Kinodaten (1/4)

Zur Illustration von Anfragen verwenden wir eine kleine Datenbank mit Kinodaten, bestehend aus

- einer Tabelle *Orte*, die Informationen über Kinos (Kino, Adresse, Telefonnummer) enthält.
- einer Tabelle *Filme*, die Informationen über Filme enthält (Titel, Regie, Schauspieler)
- einer Tabelle *Programm*, die Informationen zum aktuellen Kinoprogramm enthält (Kino, Titel, Zeit)

Material zu Abschnitt 5.6: Ein Anwendungsbereich der Logik erster Stufe: Datenbanken

Beispieldatenbank mit Kinodaten (2/4)

Orte-Tabelle:

Kino	Adresse	Telefon
Babylon	Dresdner Str. 2	61609693
Casablanca	Friedenstr. 12	6775752
Cinestar Cubix Alexanderplatz	Rathausstr. 1	2576110
Die Kurbel	Giesebrechtstr. 4	88915998
Filmpalast Berlin	Kurfürstendamm 225	8838551
International	Karl-Marx-Allee 33	24756011
Kino in der Kulturbrauerei	Schönhauser Allee 36	44354422
Moviemento	Kottbusser Damm 22	6924785

Material zu Abschnitt 5.6: Ein Anwendungsbereich der Logik erster Stufe: Datenbanken

Beispieldatenbank mit Kinodaten (3/4)

Filme-Tabelle:

Titel	Regie	Schauspieler
Capote	Bennet Miller	Philip Seymour Hoffman
Capote	Bennet Miller	Catherine Keener
Das Leben der Anderen	F. Henkel von Donnersmarck	Martina Gedeck
Das Leben der Anderen	F. Henkel von Donnersmarck	Ulrich Tukur
Der ewige Gärtner	Fernando Meirelles	Ralph Fiennes
Der ewige Gärtner	Fernando Meirelles	Rachel Weisz
Good Night and Good Luck	George Clooney	David Strathairn
Good Night and Good Luck	George Clooney	Patricia Clarkson
Knallhart	Detlev Buck	Jenny Elvers
Knallhart	Detlev Buck	Jan Henrik Stahlberg
Raumpatrouille Orion – Rücksturz ins Kino	Michael Braun	Dietmar Schönherr
Raumpatrouille Orion – Rücksturz ins Kino	Theo Mezger	Dietmar Schönherr
Raumpatrouille Orion – Rücksturz ins Kino	Michael Braun	Eva Pflug
Raumpatrouille Orion – Rücksturz ins Kino	Theo Mezger	Eva Pflug
Raumpatrouille Orion – Rücksturz ins Kino	Michael Braun	Wolfgang Völz
Raumpatrouille Orion – Rücksturz ins Kino	Theo Mezger	Wolfgang Völz
Requiem	Hans-Christian Schmid	Sandra Hüller
Sommer vorm Balkon	Andreas Dresen	Nadja Uhl
Sommer vorm Balkon	Andreas Dresen	Inka Friedrich
Sommer vorm Balkon	Andreas Dresen	Andreas Schmidt
Syriana	Stephen Gaghan	George Clooney
Syriana	Stephen Gaghan	Matt Damon
V wie Vendetta	James McTeigue	Natalie Portman
Walk the Line	James Mangold	Joaquin Phoenix
Walk the Line	James Mangold	Reese Witherspoon

Material zu Abschnitt 5.6: Ein Anwendungsbereich der Logik erster Stufe: Datenbanken

Beispieldatenbank mit Kinodaten (4/4)

Programm-Tabelle:

Kino	Titel	Zeit
Babylon	Capote	17:00
Babylon	Capote	19:30
Kino in der Kulturbrauerei	Capote	17:30
Kino in der Kulturbrauerei	Capote	20:15
International	Das Leben der Anderen	14:30
International	Das Leben der Anderen	17:30
International	Das Leben der Anderen	20:30
Filmpalast Berlin	Good Night and Good Luck	15:30
Filmpalast Berlin	Good Night and Good Luck	17:45
Filmpalast Berlin	Good Night and Good Luck	20:00
Kino in der Kulturbrauerei	Good Night and Good Luck	18:00
Kino in der Kulturbrauerei	Good Night and Good Luck	20:30
Kino in der Kulturbrauerei	Good Night and Good Luck	22:45
Babylon	Sommer vorm Balkon	21:45
Kino in der Kulturbrauerei	Sommer vorm Balkon	21:45
Filmmuseum Potsdam	Raumpatrouille Orion – Rücksturz ins Kino	22:00

Für eine geeignete Signatur σ_{Kino} können wir diese Datenbank durch eine σ_{Kino}-Struktur $\mathfrak{A}_{Kino}$ folgendermaßen modellieren:

Die Signatur σ_{Kino} besteht aus

- einem 3-stelligen Relationssymbol Orte
- - - - - - - - - - - - - - - - - Filme
- - - - - - - - - - - - - - - - - Programm
- Konstantensymbolen 'Babylon', 'Casablanca', ..., 'Capote', 'Das Leben der Anderen', ... usw.
  — dh für jeden Eintrag c in der Beispiel-datenbank gibt es ein Konstantensymbol 'c'

Die σ_{Kino}-Struktur $\mathfrak{A}_{Kino}$ hat das Universum

$$A_{Kino} := \{ \text{Babylon, Casablanca, Cinestar Cubix am Alexanderplatz}, \ldots, \text{Dresdner Str. 2}, \text{Friedenstr. 12}, \ldots, 61609693, \ldots, \text{Capote}, \ldots, 22{:}00 \},$$

die drei 3-stelligen Relationen

$\dot{Orte}^{\mathfrak{A}_{Kino}} := \{$ (Babylon, Dresdner Str. 2, 61609693),
(Casablanca, Friedenstr. 12, 6775752),
... ,
(Moviemento, Kottbusser Damm 22, 6924785) $\}$

$\dot{Filme}^{\mathfrak{A}_{Kino}} := \{$ (Capote, Bennet Miller, Philip Seymour Hoffman),
(Capote, Bennet Miller, Catherine Keener),
....
(Walk the Line, James Mangold, Reese Witherspoon) $\}$

$\dot{Programm}^{\mathfrak{A}_{Kino}} := \{$ (Babylon, Capote, 17:00),
(Babylon, Capote, 19:30),
(Kino in der Kulturbrauerei, Capote, 17:30),
... $\}$

sowie für jedes in σ_{Kino} vorkommende Konstantensymbol $'\dot{c}'$ die Konstante $'\dot{c}'^{\mathfrak{A}_{Kino}} := c$

(d.h. $'\text{Babylon}'^{\mathfrak{A}_{Kino}}$ = Babylon,
$'\text{Capote}'^{\mathfrak{A}_{Kino}}$ = Capote,
$'\text{George Clooney}'^{\mathfrak{A}_{Kino}}$ = George Clooney usw).

Anfragen an die Kinodatenbank lassen sich auf unterschiedliche Art formulieren:

Beispiel 5.35

Eine Anfrage an unsere Kinodatenbank:

"Gib die Titel aller Filme aus, die um 20:30 Uhr laufen."

In der Datenbankanfragesprache SQL lässt sich dies folgendermaßen formulieren:

```
SELECT Titel
FROM Programm
WHERE Zeit = '20:30'
```

Dieselbe Anfrage lässt sich auch durch die folgende Formel der Logik erster Stufe beschreiben:

$$\varphi_{\text{Filme um 20:30 Uhr}}(x_T) := \exists x_K \; \text{Programm}(x_K, x_T, \text{'20:30'})$$

Notation 5.36

Sei σ eine Signatur und seien $x_1, \dots, x_n$ Variablen.

- Die Notation $\varphi(x_1, \dots, x_n)$ deutet an, dass φ eine $FO[\sigma]$-Formel mit $frei(\varphi) = \{x_1, \dots, x_n\}$ ist, d.h., dass $x_1, \dots, x_n$ diejenigen Variablen sind, die in φ frei vorkommen.

- Ist $\varphi(x_1, \dots, x_n)$ eine $FO[\sigma]$-Formel, ist $\mathfrak{A}$ eine σ-Struktur und sind $a_1, \dots, a_n \in A$ Elemente im Universum von $\mathfrak{A}$, so schreiben wir
$$\mathfrak{A} \models \varphi[a_1, \dots, a_n],$$
um auszudrücken, dass für die Belegung $\beta : \{x_1, \dots, x_n\} \to A$ mit $\beta(x_1) = a_1, \dots, \beta(x_n) = a_n$ gilt:
$$(\mathfrak{A}, \beta) \models \varphi.$$

Definition 5.37

Sei σ eine Signatur, $\varphi(x_1, \dots, x_n)$ eine $FO[\sigma]$-Formel und $\mathfrak{A}$ eine σ-Struktur.

Die von φ in $\mathfrak{A}$ definierte n-stellige Relation ist

$$\varphi(\mathfrak{A}) := \{ (a_1, \dots, a_n) \in A^n : \mathfrak{A} \models \varphi[a_1, \dots, a_n] \}.$$

Beispiel 5.38:

Die $FO[\sigma_{Kino}]$-Formel $\varphi_{\text{Filme um 20:30 Uhr}}(x_T)$ (aus Bsp 5.35) definiert in unserer Kinodatenbank $\mathfrak{A}_{Kino}$ die 1-stellige Relation

$$\varphi_{\text{Filme um 20:30 Uhr}}(\mathfrak{A}_{Kino}) = \{ (\text{Das Leben der Anderen}), (\text{Good Night and Good Luck}) \}$$

Darstellung als Tabelle:

Filme um 20:30 Uhr:

Titel
Das Leben der Anderen
Good Night and Good Luck

Beispiel 5.39

Die Anfrage

„Gib Name und Adresse aller Kinos aus, in denen ein Film läuft, in dem George Clooney mitspielt oder Regie geführt hat"

lässt sich folgendermaßen formulieren:

In SQL:

```
SELECT  Orte.Kino,  Orte.Adresse
FROM    Orte, Filme, Programm
WHERE   Orte.Kino = Programm.Kino  AND
        Filme.Titel = Programm.Titel  AND
        (Filme.Schauspieler = 'George Clooney'  OR
         Filme.Regie = 'George Clooney')
```

In Logik erster Stufe:

$$\varphi_{\text{Kinos mit George Clooney}}(x_K, x_A) :=$$

$$\exists x_{Tel}\, \exists x_T\, \exists x_Z \Big(\big(\mathrm{Orte}(x_K, x_A, x_{Tel}) \wedge \mathrm{Programm}(x_K, x_T, x_Z) \big) \wedge \big(\exists x_R\, \mathrm{Filme}(x_T, x_R, \text{'George Clooney'}) \vee \exists x_S\, \mathrm{Filme}(x_T, \text{'George Clooney'}, x_S) \big) \Big)$$

In unserer konkreten Kinodatenbank $\mathfrak{A}_{Kino}$ liefert diese Formel die 2-stellige Relation

$$\varphi_{\text{Kinos mit George Clooney}}(\mathfrak{A}_{Kino}) =$$
$$\{ (\text{Filmpalast Berlin, Kurfürstendamm 225}), (\text{Kino in der Kulturbrauerei, Schönhauser Allee 36}) \}$$

Darstellung als Tabelle:

Kinos mit George Clooney:

Kino	Adresse
Filmpalast Berlin	Kurfürstendamm 225
Kino in der Kulturbrauerei	Schönhauser Allee 36

Details zum Thema Datenbanken und Datenbankanfragesprachen können Sie in den Vorlesungen "Datenbanksysteme I und II" und "Logik und Datenbanken" kennenlernen.