

# Prädikatenlogik

- Syntax
- Semantik
- Formeln, Modelle, Tautologien und Anwendungen
- Entscheidbarkeitsfragen
- Kalküle und Tableaux
- Normalformen und Resolution

# Prädikatenlogik

- **Syntax**

## Von der Aussagenlogik zur Prädikatenlogik (1)

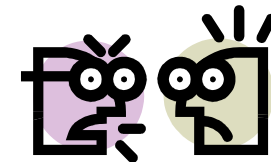
In der **Aussagenlogik** ...

betrachten wir **globale Aussagen über ganze Systeme**.

Aussagen sind **je nach betrachtetem System** (konkretes System bzw. abstraktes System = Belegung der Aussagevariablen) **wahr oder falsch**.  
**Keine** logische Behandlung **interner Details** der Systeme.

Was ist aber mit (logischen?) Argumentationen wie ...

„**Wer jemanden** anmeckert, fliegt raus!“ –  
„Aber **Anna** hat **mich** doch gerade angemekert  
und ist nicht rausgeflogen!“



„**Gibt es** Adverbien **die** mit S anfangen?“ –  
„Sicher!“

..., bei denen es **im betrachteten System** um Eigenschaften oder Beziehungen **einzelner oder aller Individuen einer Population** geht?

## Von der Aussagenlogik zur Prädikatenlogik (2)

**Prädikatenlogik:**

die Welt, die ganzen Zahlen, ...

Wir **schauen in die Systeme hinein** und befassen uns mit

- **Objekten** innerhalb eines Systems sowie ihren möglichen
- **Eigenschaften** und
- **Beziehungen** zueinander.

**Aussagen** sprechen vom **Zutreffen von Eigenschaften** oder **Beziehungen** auf **einzelne** oder **alle Objekte** innerhalb des Systems.

Aussagen sind in der Regel **Ausnahmen?** **je nach**

- betrachtetem **System und** den
- bezeichneten **Objekten** und **Eigenschaften** oder **Beziehungen**

**wahr oder falsch.**

## Der Weg zur PL-Syntax und -Semantik (1)

**Syntax:** AL + zusätzliche **Ausdrucks Mittel zur Benennung** von

- bestimmten oder beliebigen **Objekten**:  
→ Eigennamen (*Konstanten, Variablen*), Terme (auch mit *Funktionen*)
- **Eigenschaften** von Objekten oder **Beziehungen** zwischen mehreren Objekten  
→ *Prädikate, Relationen*

**Syntaxbeispiel:**

sprachlich: Zu jedem Objekt existiert ein echt größeres.

symbolisch:  $\forall x \exists y G(y, x)$

## Der Weg zur PL-Syntax und -Semantik (2)

**Semantik:** Interpretation der Objekt- und der Eigenschaften-Namen in einem konkreten System.

### Semantikbeispiele:

Die obige Aussage ist

- **wahr**, wenn 

Objekte := Intervalle $[r,s]$ in den reellen Zahlen, mit $[a,b]$ größer als $[c,d] \iff b - a > d - c$ ;
--
- **falsch**, wenn 

Objekte := Münzen in meiner Geldbörse, Größe $\cong$ Durchmesser
---

## „Alphabet“ der Prädikatenlogik erster Stufe („PL1“)

Wir definieren das **Alphabet**, die **Terme** (für Objekte) und dann **PL1-Formeln**.

Das **Alphabet** der Prädikatenlogik erster Stufe PL1 umfasst

- eine abzählb. Menge  $VS$  von **(Objekt-) Variablensymbolen**, hier  $x_1, x_2, \dots$ 
  - informell  $u, v, w, x, y, z, \textit{Mitarbeiter}$  usw.
- eine abzählb. Menge  $KS$  von **(Objekt-) Konstantensymbolen**, hier  $a_1, a_2, \dots$ 
  - informell  $a, b, c, \dots, \textit{Besitzer}$  usw.
- eine abzählbare Menge  $FS$  von **Funktionssymbolen**, hier  $f_1, f_2, \dots$ 
  - informell  $f, g, h, \dots, \textit{Vorgesetzte}(r)$ , usw. –
  - mitsamt Abbildung (F-)Stelligkeit :  $FS \rightarrow \mathbb{N}$
- eine abzählbare Menge  $PS$  von **Prädikat-(Relations-)symbolen**, hier  $P_1, P_2, \dots$ 
  - informell  $P, Q, R, \dots, \textit{arbeitet_für}$ , usw. –
  - mitsamt Abbildung (P-)Stelligkeit :  $PS \rightarrow \mathbb{N}$

## Alphabet von PL1 (Rest)

- **Junktoren**

$\neg \wedge \vee \rightarrow \leftrightarrow$

- **Quantoren**

$\forall$  *für alle ... gilt ...*

**Allquantor, Generalisierung**

$\exists$  *es existiert (mindestens) ein ... so dass ...*

**Existenzquantor, Partikularisierung**

- **syntaktische Hilfssymbole** Klammern ( ) und Komma , .



auch mal als { } oder [ ]



## Sprache der Prädikatenlogik PL1: Terme

Mit **Termen** benennen wir **Objekte**.

Die PL1-Terme sind **induktiv** so definiert:

- Alle (Objekt-) **Konstanten** und **Variablen** sind **Terme**;
- Ist  $f$  ein  $k$ -stelliges **Funktionssymbol** ( $k \geq 1$ ) und sind  $t_1, \dots, t_k$  **Terme**, so ist  $f(t_1, \dots, t_k)$  ein **Term**.

Alternativer PL1-Dialekt:

*Objektkonstanten  $\cong$  „nullstellige“ Funktionen*

## Sprache der Prädikatenlogik PL1: Formeln

Die Aussagen der PL1 heißen **(PL1)-Formeln**.

PL1-Formeln sind **induktiv** so definiert:

- Ist  $P$  ein  $k$ -stelliges Relationssymbol ( $k \geq 1$ ) und sind  $t_1, \dots, t_k$  Terme, so ist  $P(t_1, \dots, t_k)$  eine (**atomare**) Formel;

alternativer PL1-Dialekt: *nullst. Rel.n  $\equiv$  „Aussagenvariablen“*

- Sind  $F$  und  $G$  Formeln, dann auch  $\neg F$ ,  $(F \wedge G)$ ,  $(F \vee G)$ ,  $(F \rightarrow G)$ , und  $(F \leftrightarrow G)$ .
- Ist  $F$  eine Formel und  $x$  eine Variable, so sind  $\forall xF$  und  $\exists xF$  Formeln.

**komplexe** Formeln: nicht atomar,

**Teilformeln, Teilterme** von Formeln  
**Induktion/Rekursion** über Formelaufbau  
**Formel-** und/oder **Termbäume** } analog  
 AL

evtl. Klammern  
 wg. Lesbarkeit, z.B.

$$\forall x P(x) \rightarrow \exists y Q(x, y) \rightarrow$$

$$\forall x [P(x) \rightarrow \exists y Q(x, y)] \quad ??$$

$$\forall x [P(x)] \rightarrow \exists y [Q(x, y)] \quad !!$$

## PL1 und natürliche Sprache: Übung

Darmstadt ist eine Großstadt südlich von Frankfurt.

Keine deutsche Großstadt liegt nördlich von Flensburg.

Alle Menschen werden Brüder.

*Vorsicht - Sprachfeinheiten!*

## PL1 und natürliche Sprache: Lösung

Darmstadt ist eine Großstadt südlich von Frankfurt(M).

$da := \text{Darmstadt}$   
 $f := \text{Frankfurt(M)}$   
 $Gr(x) :\Leftrightarrow x \text{ ist Großstadt}$   
 $S(x, y) :\Leftrightarrow x \text{ liegt südlich von } y$

$Gr(da) \wedge S(da, f)$

Keine deutsche Großstadt liegt nördlich von Flensburg.

$fl := \text{Flensburg}$   
 $Gr(x) :\Leftrightarrow x \text{ ist Großstadt}$   
 $N(x, y) :\Leftrightarrow x \text{ liegt nördlich von } y$

$\neg \exists x (Gr(x) \wedge N(x, fl))$

Alle Menschen werden Brüder.

*Vorsicht - Sprachfeinheiten!*

$M(x) :\Leftrightarrow x \text{ ist Mensch.}$   
 $Br(x, y) :\Leftrightarrow x \text{ wird Bruder von } y$

$\forall x \forall y (x \neq y \wedge (M(x) \wedge M(y))) \rightarrow Br(x, y)$

*x ≠ y ?!*

*Zeit? Veränderung?*

## PL1 und natürliche Sprache: Grundmengen, Konstanten

- Wovon wird gesprochen?  
D.h. wer sind die Objekte bzw. die **Grundmengen**?
- Die hatten wir stillschweigend vorausgesetzt.
- Teils verschiedene sinnvolle Möglichkeiten:

Darmstadt ist eine Großstadt südlich von Frankfurt.

Keine deutsche Großstadt liegt nördlich von Flensburg.

→ **Grundmenge** (deutsche) (Groß-)Städte, Örtlichkeiten auf der Erde, ...

→ **Konstanten**: Darmstadt, Frankfurt, Flensburg

Alle Menschen werden Brüder.

→ **Grundmenge** Menschen, Lebewesen, Gegenstände und Lebewesen, ...

→ **Konstanten**: –

## **PL1 und natürliche Sprache: gemischte Populationen, Funktionen**

Deine Meinung interessiert mich nicht.

Grundmenge  
Konstanten  
Funktionen  
Relationen

Brunos Vater ist ein Vorgesetzter von Stefanies Mutter.

## PL1 und natürliche Sprache: gemischte Populationen, Funktionen

Deine Meinung interessiert mich nicht.

Grundmenge: Personen + Meinungen

Konstanten: *ich, du* Personen

Funktionen:  $m(x)$  Meinung von  $x$  (mit z.B.  $m(w)=w/\emptyset$  für Meinungen  $w$ .)

Relationen:  $Int(z,y)$  Meinung  $z$  interessiert Person  $y$ . (Einschränkung OK)

→  $\neg Int(m(du), ich)$

Brunos Vater ist ein Vorgesetzter von Stefanies Mutter.

Grundmenge: Personen

Konstanten: *br, st* Personen

Funktionen:  $mu(x)$  Mutter von  $x$

$va(x)$  Vater von  $x$

Relationen:  $Vor(x,y)$   $x$  ist ein Vorgesetzter von  $y$ .

→  $Vor(va(br), mu(st))$

# PL1 und natürliche Sprache – Quantorenmuster (1)

Grundmenge, Konstanten, Funktionen, Prädikate, Relationen?



Formel

- Alle lachen.
 

Grundmenge:	irgendeine nicht näher bezeichnete Personengruppe	}	$\forall x L(x)$
Prädikate:	$L(x)$ – $x$ lacht		
- (Mindestens) Einer lacht.
 

Grundmenge, Prädikat:	s.o.	}	$\exists x L(x)$
-----------------------	------	---	------------------
- Keiner lacht.
 

Grundmenge, Prädikat:	s.o.	}	$\neg \exists x L(x)$
-----------------------	------	---	-----------------------
- Alle lachen mich aus (sogar ich mich selbst).
 

Grundmenge:	s.o.,	}	$\forall x La(x,i)$
Relation:	$La(x,y)$ – $x$ lacht $y$ aus,		
Konstante:	$i$ – ich		



## PL1 und natürliche Sprache – Quantorenmuster (2)

- Alle (anderen) lachen mich aus.  
 Grundmenge, Relation, Konstante: s.o. }  $\forall x (x \neq i \rightarrow La(x,i))$

PL1 mit Identität!  $x \neq y$  bedeutet  $\neg = (x,y)$

- Alle Schadenfrohen lachen mich aus.  
 Grundmenge, Relation, Konstante: s.o.  
 + Prädikat:  $Sf(x)$  –  $x$  ist schadenfroh }  $\forall x (Sf(x) \rightarrow La(x,i))$

**Oben sehen wir ein wichtiges Grundmuster:**

**Für alle mit Eigenschaft  $E$  gilt ...  $\rightarrow$   $\forall x (E(x) \rightarrow \dots)$**

Üben: Alle Vögel fliegen hoch. (Population: Tiere)

Alle Studierenden lieben Logik. (Population: Menschen)

Etwas schwieriger:

Alle logikfeindlichen Studierenden lieben alle Fächer außer Logik.

Alle Nicht-Studenten bewundern alle Studenten, die Logik verstehen.

(Population: Menschen und Fächer, Prädikate? Relationen?)

## PL1 und natürliche Sprache – Quantorenmuster (3)

- Some like it hot.<sup>1</sup> bzw.  
Manche (hier: *mindestens eine(r)*) mögen's heiß. }  $\exists x Mh(x)$
- Manche Studierenden lieben Logik.  
Grundmenge Menschen (evtl. und Fächer),  
Prädikat:  $St$  – ist Studierende(r) } ?

### Das andere wichtige Grundmuster:

Für mindestens eines mit Eigenschaft  $E$  gilt ...  $\rightarrow \exists x (E(x) \wedge \dots)$

Also könnte –?– lauten:  $\exists x (St(x) \wedge liLo(x))$

Oder in der Population aller Menschen und Fächer  
könnte –?– lauten:  $\exists x (St(x) \wedge Li(x, lo))$

Üben: Everybody loves somebody.<sup>2</sup>

Alle Studenten verstehen mindestens ein Fach neben Logik.

PL1 mit Identität.



<sup>1</sup> Billy Wilder

<sup>2</sup> Dean Martin; Blues Brothers

## PL1 verallgemeinert AL

... d.h. wir können AL in PL1 **reproduzieren**:

1. durch Beschränkung auf **nullstellige Prädikate**(nsymbole).  
**Keine** Quantoren, Objekte (Var'en, Konst'en), Funktionen.

PL1-Theorie liefert AL-Theorie.

2. Sogar allgemeiner, nur:

**Keine** Quantoren oder Objektvariablen! (Konst'en, Funktionen, zugelassen)

$$(Q(a) \vee \neg R(f(b), c)) \wedge P(a, b) \quad \longrightarrow \quad (Q\_a \vee \neg R\_f\_b\_c) \wedge P\_a\_b$$

Wie Symbole  
aufzählen?

## PL2 verallgemeinert PL1

In PL2 **quantifiziert** man auch über **Prädikate und Funktionen**, z.B. im **Induktionsaxiom** der natürlichen Zahlen:

Sei  $P$  eine mögliche Eigenschaft natürlicher Zahlen, und

- 1 hat die Eigenschaft  $P$ , und
- mit jeder Zahl  $n$  hat auch  $n+1$  die Eigenschaft  $P$ .

Dann haben alle natürlichen Zahlen die Eigenschaft  $P$ .

$$\forall P \langle (P(1) \wedge \forall n [P(n) \rightarrow P(\text{plus}(n,1))]) \rightarrow \forall n P(n) \rangle$$

gebundene  
Prädikatvariable

**aber** in axiomatischer Mengenlehre (PL1!)  
ist beschreib- und beweisbar:

$$\forall M \langle (\text{eins} \in M \wedge \forall n [n \in \text{Nat} \cap M \rightarrow \text{plus}(n, \text{eins}) \in M]) \rightarrow \text{Nat} \subseteq M \rangle$$

## PL1-Varianten (1)

### Andere Schreibweisen

#### – Quantoren:

$\forall x F$  auch so:  $\bigwedge_x F$ ,  $\forall x: F$ ,  $(\forall x) F$

$\exists x F$  auch so:  $\bigvee_x F$ ,  $\exists x: F$ ,  $(\exists x) F$

#### – Relationen: auch Infix, Mixfix.

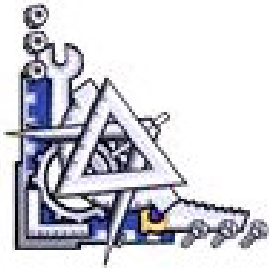
#### – Relationen- und Funktionensymbole: $f_1^k, f_2^k, \dots, P_1^k, P_2^k, \dots$ (pro Stelligkeit $k$ )

#### – Relationen- / Funktionensymbole: 0-stellige erlauben – als Ausdrucksmittel für globale (objektunabhängige) Aussagevariablen bzw. Objektkonstanten (bereits erwähnt)

## PL1-Varianten (2)

### Andere **Formel- und Bedeutungsregeln** (Auswahl)

- PL1 mit **Identität**:  
Vordefinierte zweistellige Relation =. (auch  $\equiv$  oder  $\approx$  geschrieben)
- PL1 mit **Sorten**:  
Objekte sind oft **unterschiedlicher** Art, z.B. Zahlen, Vektoren, Matrizen.  
Relationen und Funktionen sind **sortenspezifisch**, z.B. lineare Abbildung ist „Matrix mal Vektor“, Kreuzprodukt ist „Vektor mal Vektor“.  
Technik: getrennte Variablen- und Konstanten-Namen-Mengen, oder alle Quantoren an Sorten binden:  $\forall x \in M \exists y \in N P(x, y)$ .  
Sorten-Ersatz in „unsortierter“ PL1:  $\forall x (M(x) \rightarrow \exists y (N(y) \wedge P(x, y)))$
- PL1 **ohne** besondere **Funktionensymbole**:  
Berechtigung: Funktionen sind spezielle Relationen. Nachteil:  
Linkstotalität und Rechtseindeutigkeit geeignet explizit „mitzunehmen.“



## Wichtiger technischer Kleinkram Freie und gebundene Variablen (1)

<u>Begriff</u>	<u>Definition</u>	<u>Im Formelbaum</u>
<b>[Text]</b>	Ein <b>Vorkommen</b> (Exemplar) von <i>Text</i> in einer Formel. (Kann ja mehrfach vorkommen!)	
<b>Scope, Bereich</b>	In $\forall x F$ und $\exists x F$ ist <b>F</b> der Scope vom äußersten $[\forall x]$ / $[\exists x]$ .	der ganze Ast unter dem Quantorknoten
<b>Quantorvariable (-nvorkommen)</b>	das $[x]$ in $\forall x F$ und $\exists x F$	

## Freie und gebundene Variablen (2)

<u>Begriff</u>	<u>Definition</u>	<u>Im Formelbaum</u>
<b>frei</b>	<p><math>[x]</math> ist frei in <math>F</math>, wenn es Teilterm von <math>F</math> ist und nicht im Scope eines <math>[\forall x]/[\exists x]</math> steht.</p> <p>also <i>nicht</i> Q.-var. in <math>[\forall x]/[\exists x]</math>.</p>	
<b>gebunden durch</b>	<p><math>[x]</math> ist gebunden durch <math>[\forall x]/[\exists x]</math>, wenn es im Scope des <math>[\forall x]/[\exists x]</math> frei vorkommt.</p>	Das unterste $[\forall x]/[\exists x]$ oberhalb $[x]$ bindet $[x]$ .
<b>geschlossene Formel</b>	<p>... hat <b>keine freien</b> Variablen(vorkommen), auch <u>Satz</u> oder <u>Aussage</u> genannt</p> <p>Theorem?</p> <p>Aussagenlogik?</p>	



# Freie und gebundene Variablen im Baum

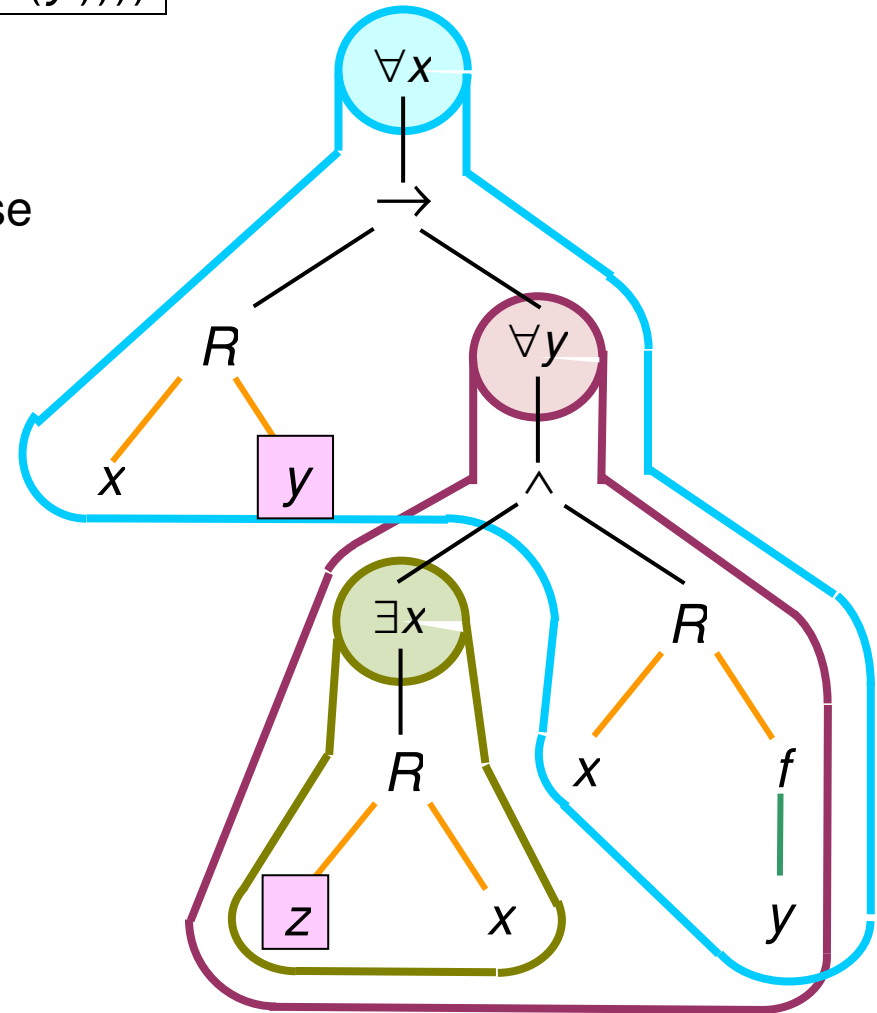
Beispiel :  $\forall x(R(x,y) \rightarrow \forall y(\exists x R(z,x) \wedge R(x,f(y))))$

Unterhalb  $[\forall x]$  sind alle  $[x]$  gebunden: Scope!

**Allerdings** wird die Bindung durch  $[\forall x]$  teilweise **abgelöst** von der Bindung durch  $[\exists x]$ .

Ein  $[y]$  ist frei, ein  $[y]$  ist Quantorvariable (das in  $[\forall y]$ ), ein  $[y]$  ist gebunden.

Alle  $[z]$  sind frei.



## Legende

- x freies Vorkommen
- Aufbau komplexer Formel
- Aufbau atomarer Formeln
- Aufbau von Termen

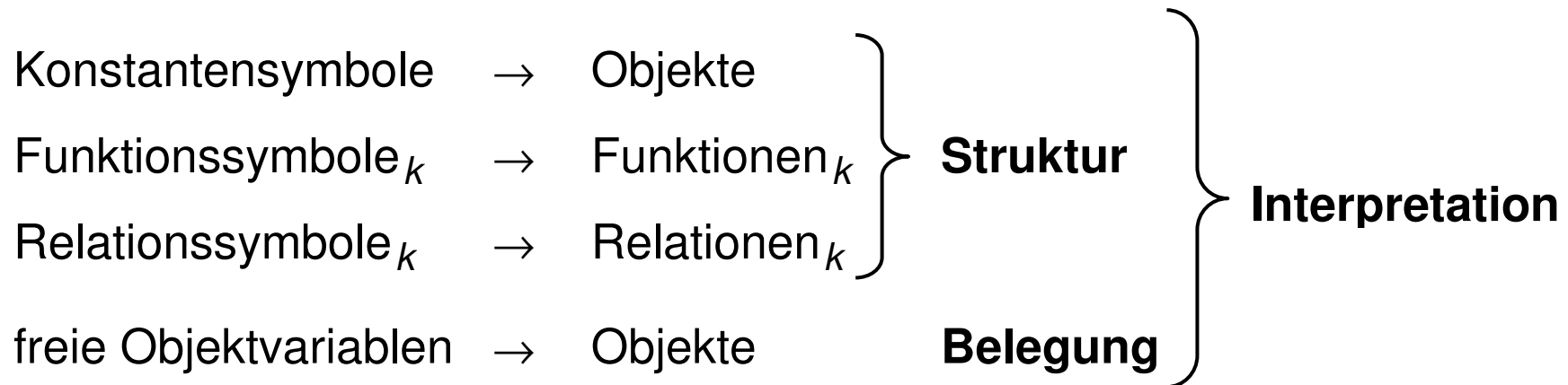
Ü45 →

# Prädikatenlogik

- Syntax
- **Semantik**

## Interpretationen: Zweistufigkeit

### Zuordnungen ...



**Gebundene** Variablenvorkommen werden dabei

- „**nicht notwendig interpretiert**“,
- und wenn ja, wird diese **Interpretation** ohnehin **ignoriert**,
- sondern bei der **Auswertung** verwendet und dabei systematisch „**mit allen möglichen Objekten**“ belegt!

## Interpretationen: Strukturen

**Struktur**  $A=(U, I_A)$  :

- $U =$  **nicht leere Menge** <sup>(1)</sup> von Objekten (**Grundmenge**, Universum)
- $I_A =$  **Interpretation der Strukturparameter** – Tripel  $(I_K, I_F, I_P)$  von Abbildungen

$$I_K : KS_A \rightarrow U, KS_A \subseteq KS$$

von (einigen) **Konstantensymbolen** auf Objekte in  $U$ ;

$$I_F : FS_A \rightarrow \bigcup_{k=1}^{\infty} F(U^k, U), FS_A \subseteq FS$$

von **Funktionssymbolen** auf ein- oder mehrstellige Funktionen auf  $U$ , wobei  $k$ -stellige Symbole  $\mapsto$   $k$ -stellige Funktionen;

$$I_P : PS_A \rightarrow \mathbf{P}(\bigcup_{k=0}^{\infty} U^k), PS_A \subseteq PS$$

von **Prädikat-** und **Relationssymbolen** auf ein- oder mehrstellige Relationen über  $U$ , wobei  $k$ -stellige Symbole  $\mapsto$   $k$ -stellige Relationen.

<sup>(1)</sup> **denkbar: andersartige Variante, LT-Semantik,**  
 evtl. leerer Träger,  $U = \emptyset$

$$(\forall x F) \rightarrow (\exists x F) ?$$

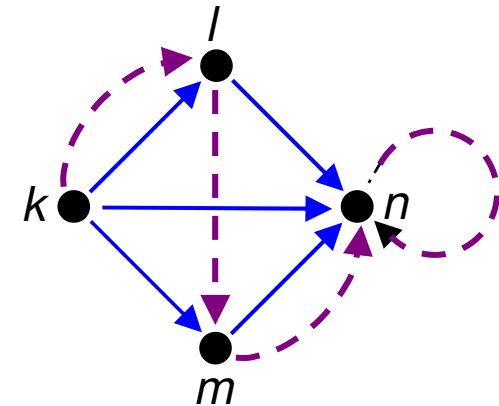
## Eine Beispiel-Struktur A\_Bsp

### Alphabet:

**Konstantensymbole**  $KS$ :  $c$

**Funktionssymbole**  $FS$ :  $f$  (1-stellig)

**Relationssymbole**  $RS$ :  $P$  (2-stellig)



### Struktur:

**Grundmenge:**

$$U := \{ k, l, m, n \}$$

**Interpretation:**

**Konstanten:**

$$I_K : c \mapsto l$$

sonst. **Funktionen:**

$$I_F : f, \text{ gegeben durch } x \dashrightarrow f(x)$$

**Relationen:**

$$I_P(P)(x, y) := x=y \text{ oder } x \longrightarrow y$$

In einer Struktur können bereits „syntaktisch konstante“ Terme und Formeln ausgewertet werden.

## Struktur und Auswertung konstanter Terme

Eine Struktur  $A = (U, (I_K, I_F, I_R))$  heißt

**passend** oder **ausreichend** für PL1-Term  $\tau$ ,

wenn sie **alle Konstanten-, Funktions-, und Prädikatsymbole** in  $\tau$  interpretiert.

In diesem Falle ...

**Auswertung von (konstanten) Termen** in  $A$  (rekursiv):

- $c$  Konstante:  $I(c) := I_K(c)$
- $f$  Funktionssymbol und  $t_1, t_2, \dots, t_k$  Terme:  
 $I(f(t_1, \dots, t_k)) := I_F(f)(I_A(t_1), \dots, I_A(t_k))$

Ü46 →

## Interpretation – Belegung und Auswertung von Termen

**Variablen-Interpretation** (oder **Belegung**) in einer Struktur  $A=(U, I_A)$ :

Abbildung  $I_V : VS_{I_V} \rightarrow U$ ,  $VS_{I_V} \subseteq VS$ ,

von (einigen) Variablen(symbolen) auf Objekte in  $U$ .

bzw.  $(U, I_K, I_F, I_R, I_V)$

Ein Paar  $I = (A, I_V) = (\text{Struktur}, \text{Belegung})$ , eine **Interpretation**, heißt **passend** oder **ausreichend** für eine PL1-Formel  $\varphi$ , wenn es alle

- Konstanten-, Funktions-, und Prädikatsymbole,
- sowie alle **freien** Variablen in  $\varphi$  interpretiert.

In diesem Falle ist  $A$  ausreichend für alle Terme in  $\varphi$  ohne gebundene Variablen, und die **Formel**  $\varphi$  kann **ausgewertet** werden.

**Auswertung von „beliebigen“ Termen:**  $I(\tau)$  wie  $I_A(\tau)$  + zusätzlich ...

- $x$  Variable (genügt:  $[x]$  freies Variablenvorkommen):  $I(x) := I_V(x)$

## Belegung und Formel-Auswertung (1)

**Auswertung** von **Formeln** unter passendem  $I$  (rekursiv):

- **atomare Formeln:**  $I$  sei ausreichend für  $P(t_1, \dots, t_k)$ , wobei  $P$  Prädikatsymbol und  $t_1, t_2, \dots, t_k$  Terme:

$$I(P(t_1, \dots, t_k)) := \begin{cases} W & \text{wenn } (I(t_1), \dots, I(t_k)) \in I_P(P) \\ F & \text{sonst} \end{cases}$$

- „**Junktorenformeln**“:  $I$  sei ausreichend für Formeln  $\varphi, \psi$ .  
 $I(\neg\varphi), I(\varphi \wedge \psi), I(\varphi \vee \psi), I(\varphi \rightarrow \psi), I(\varphi \leftrightarrow \psi)$  ergeben sich aus  $I(\varphi), I(\psi)$  wie in der **Aussagenlogik**.



## Belegung und Formel-Auswertung (2)

„Quantorenformeln“:  $I$  sei ausreichend für Formel  $\forall x\varphi$

- $I(\forall x\varphi) := \begin{cases} W & \text{wenn für alle } a \in U \text{ gilt: } I_{x:=a}(\varphi) = W \\ F & \text{sonst} \end{cases}$

Belegung außer-  
halb  $x$  festhalten.  
Mit  $x$  das ganze  $U$   
durchprüfen.

- $I(\exists x\varphi) := \begin{cases} W & \text{wenn es } a \in U \text{ gibt mit: } I_{x:=a}(\varphi) = W \\ F & \text{sonst} \end{cases}$

Wie  $I$ ,  
aber  $I_V(x) := a$

Unterschied  $x:=a$  /  $[x/a]$  ? :  
 $x:=a$  wertemäßig, nicht text-  
lich; auch für „namenlose“  $a$ ;  
betr. nur freie Vorkommen!

## Die zwei Richtungen der PL1-Formel-Auswertung

Die Interpretation und die Auswertung

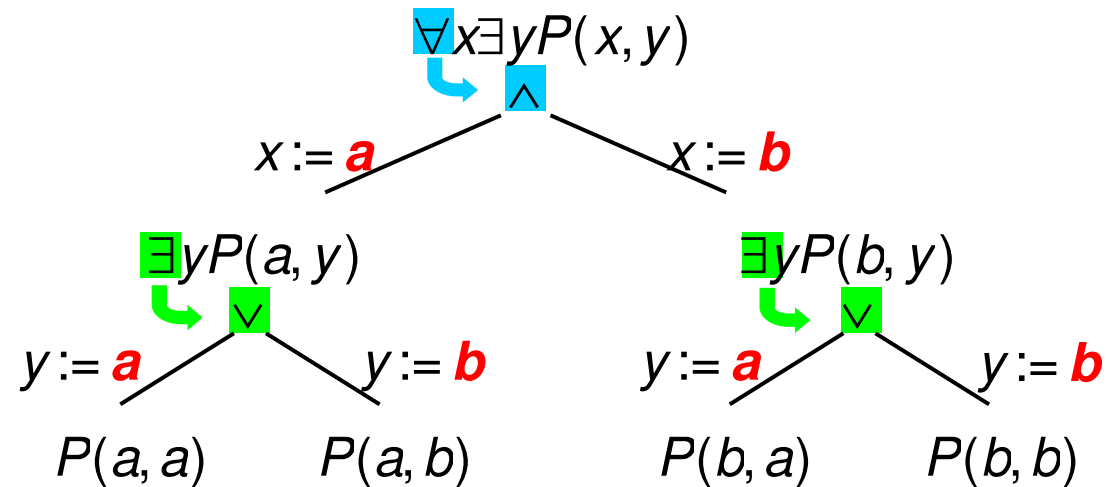
- verlaufen also in jeder quantorfreien Formel von innen nach außen (im Formelbaum: von den Blättern zur Wurzel).

Für jede quantisierte Teilformel

- werden aber von außen nach innen (im Formelbaum: von der Wurzel weg zu den Blättern) Auswertungen für im Prinzip alle Belegungen der quantifizierten Variablen mit Werten aus ganz  $U$  angestoßen, was zu einer Vielzahl (meist unendlichen Zahl) von Auswertungen quantorfreier Formeln führt.

## Ein Beispiel für die PL1-Formel-Auswertung

Sei  $U = \{a, b\}$ .



... entspricht also:

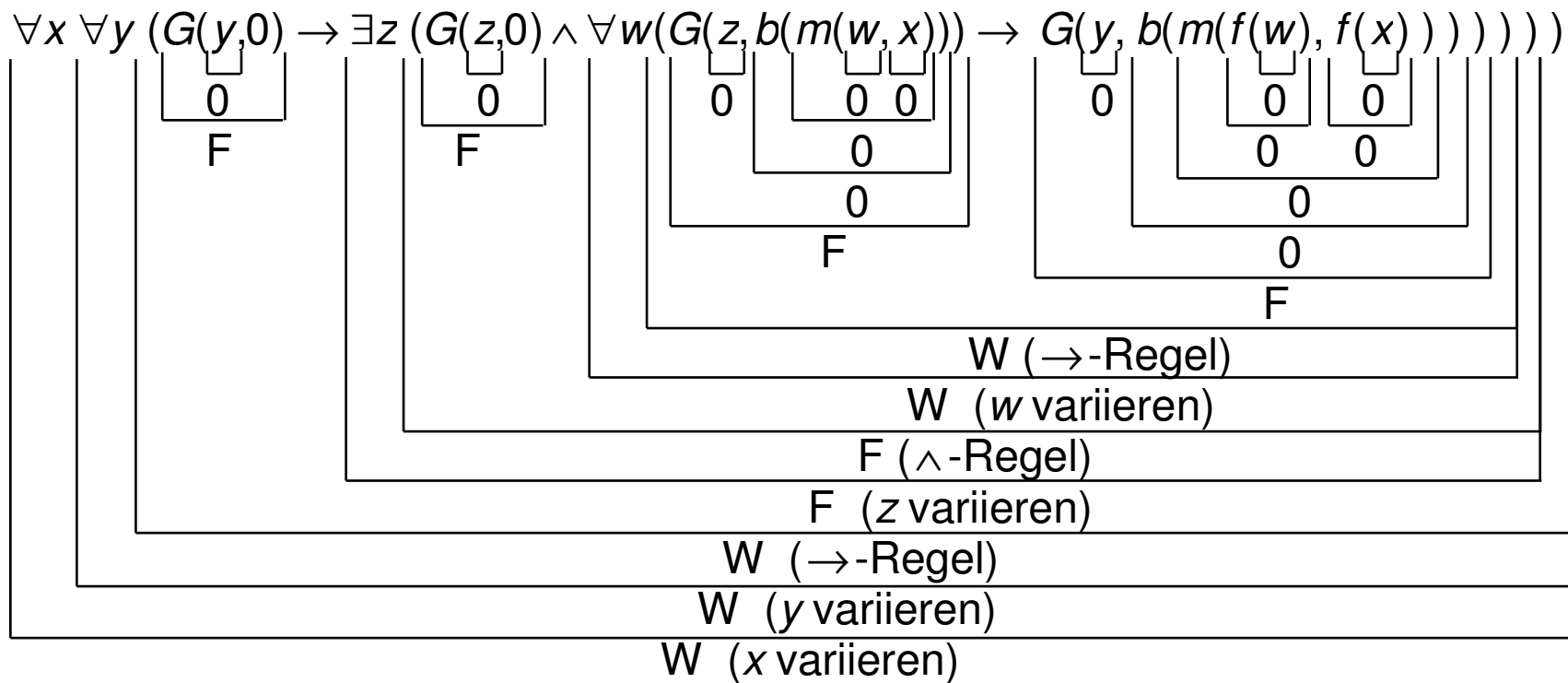
$$[ ( P(a,a) \vee P(a,b) ) \wedge ( P(b,a) \vee P(b,b) ) ]$$

# Formel-Auswertung unter verschiedenen Interpretationen (1)

Formel  $\boxed{\forall x \forall y (G(y, n) \rightarrow \exists z (G(z, n) \wedge \forall w (G(z, b(m(w, x))) \rightarrow G(y, b(m(f(w), f(x)))))) )}$  mit ...

- Objektkonstante  $n$
- Funktionssymbolen  $m$  (2-stellig),  $b, f$  (1-stellig)
- Objektvariablen  $w, x, y, z$
- Relationssymbolen  $G$  (2-stellig)

1. Interpretation  $U = \{0\}$ ,  $n, w, x, y, z \mapsto 0$ ,  $b, f, m$  auch klar (wieso?),  $G \mapsto \{ \}$



## Formel-Auswertung unter verschiedenen Interpretationen (2)

Formel  $\boxed{\forall x \forall y (G(y, n) \rightarrow \exists z (G(z, n) \wedge \forall w (G(z, b(m(w, x))) \rightarrow G(y, b(m(f(w), f(x))))))})}$  mit ...

- Objektkonstante  $n$
- Funktionssymbolen  $m$  (2-stellig),  $b, f$  (1-stellig)
- Objektvariablen  $w, x, y, z$
- Relationssymbolen  $G$  (2-stellig)

### 2. Interpretation

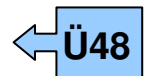
$U = \mathbb{R}$ ,  $n, \underbrace{w, x, y, z}_{\text{fiktiv}} \mapsto 0$ ,  $b(x) := |x|$ ,  $f(x) := x^2$ ,  $m(x, y) := x - y$ ,  $G(x, y) := x > y$

Etwas anders schreiben:

$\forall x_0 \forall \varepsilon > 0 \exists \delta > 0 : \forall x (|x - x_0| < \delta \rightarrow |x^2 - x_0^2| < \varepsilon)$

$\Rightarrow f(x) = x^2$  stetig?

Wahr!



# Prädikatenlogik

- Syntax
- Semantik
- **Formeln, Modelle, Tautologien und Anwendungen**

## Semantische Begriffe (1)

Eine zu einer Formel  $\varphi$  passende Interpretation  $I = (A, I_V)$  heißt

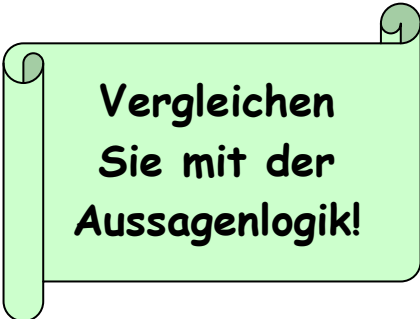
**Modell von  $\varphi$**   $:\Leftrightarrow I(\varphi) = W$

Schreibweise:  $I \models \varphi$

Redeweise:  $\varphi$  **gilt unter  $I$** , bzw.  **$I$  erfüllt  $\varphi$** .

Die Formel  $\varphi$  heißt

- **erfüllbar**, wenn sie mindestens **ein** Modell hat (**sonst: unerfüllbar**);
- **allgemeingültig** (logisch wahr,  $\models \varphi$ )  
wenn **jede** passende Interpretation Modell ist (**sonst: widerlegbar**);
- **gültig in der Struktur  $A$**  ( $A \models \varphi$ ),  
wenn für jede passende Belegung  $I_V$  die Interpretation  $I = (A, I_V)$  Modell ist.  
Beispiel:  $a + b = b + a$  „gilt in  $\mathbb{N}_0$ .“



**Vergleichen  
Sie mit der  
Aussagenlogik!**

## Semantische Begriffe (2)

Zwei Formeln  $\varphi$  und  $\psi$  heißen

- (logisch, semantisch) **äquivalent**,  $\boxed{\varphi \equiv \psi}$ ,  
wenn unter jeder passenden Interpretation  $I$  gilt:  $I(\varphi) = I(\psi)$ .

Eine Formel  $\varphi$

- **folgt** (logisch, semantisch) aus einer Formel  $\sigma$  (Formelmenge  $M$ ),  
geschrieben:  $\boxed{\sigma \models \varphi}$  ( $\boxed{M \models \varphi}$ ),  
wenn jedes Modell von  $\sigma$  (bzw.  $M$ ) auch Modell von  $\varphi$  ist.



## Wichtige Äquivalenzen (1)

Erinnerung:

Zwei Formeln  $\varphi$  und  $\psi$  heißen (logisch, semantisch) **äquivalent**,  $\varphi \equiv \psi$ , wenn unter jeder zu beiden passenden Interpretation  $I$  gilt:  $I(\varphi) = I(\psi)$ .

**Dualität** von  $\forall$  und  $\exists$

- $\neg \forall x \varphi \equiv \exists x \neg \varphi$
- $\neg \exists x \varphi \equiv \forall x \neg \varphi$

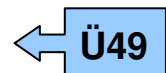
**Vertauschung** der Quantoren-Reihenfolge

nur  $\forall/\forall$  oder  $\exists/\exists$   
- nicht  $\forall/\exists$  !

- $\forall x \forall y \varphi \equiv \forall y \forall x \varphi$
- $\exists x \exists y \varphi \equiv \exists y \exists x \varphi$

**Quantoreinführung** und **-beseitigung** – Voraussetzung:  $x$  **nicht frei** in  $\varphi$

- $\varphi \equiv \forall x \varphi$
- $\varphi \equiv \exists x \varphi$



## Wichtige Äquivalenzen (2)

(Leere) **Scope-Erweiterung** (auf  $\psi$ ) – Voraussetzung:  $x$  **nicht frei** in  $\psi$ .

- $\forall x \varphi \wedge \psi \equiv \forall x (\varphi \wedge \psi)$
- $\exists x \varphi \vee \psi \equiv \exists x (\varphi \vee \psi)$
- $\forall x \varphi \vee \psi \equiv \forall x (\varphi \vee \psi)$
- $\exists x \varphi \wedge \psi \equiv \exists x (\varphi \wedge \psi)$
- $\psi \rightarrow \forall x \varphi \equiv \forall x (\psi \rightarrow \varphi)$
- $\psi \rightarrow \exists x \varphi \equiv \exists x (\psi \rightarrow \varphi)$

aber Achtung:  $\forall x \varphi \rightarrow \psi \not\equiv \forall x (\varphi \rightarrow \psi)$  !

### Distributivität

- $\forall x \varphi \wedge \forall x \psi \equiv \forall x (\varphi \wedge \psi)$
- $\exists x \varphi \vee \exists x \psi \equiv \exists x (\varphi \vee \psi)$

Nicht „über Kreuz“, d.h.  $\forall + \vee$  /  $\exists + \wedge$  !

Gegenbeispiele in  $\mathbb{N}_0$ ?

Z.B. mit  $x > 0$ ,  $0 > 0$ ,  $x$  un/gerade,  $x = 0/1$

## „AL-Substitution“ in PL1

### Satz über AL-Tautologien in PL1:

Werden in einer AL-Tautologie (bzw. in einer unerfüllbaren AL-Formel) alle Vorkommen jeder Aussagevariablen jeweils durch die gleiche PL1-Formel ersetzt, so ist die entstehende PL1-Formel ebenfalls eine Tautologie (bzw. unerfüllbar).

Dies ist eine Abart der **AL-Substitution**, formale Definition wie dort!

## Ersetzungssatz in PL1

### Ersetzungssatz (Satz über die äquivalente Ersetzung):

Werden in einer Formel  $\varphi$  ein oder mehrere Vorkommen einer Teilformel  $\psi$  durch eine zu  $\psi$  äquivalente Formel  $\rho$  ersetzt, so ist die entstehende Formel zu  $\varphi$  äquivalent.

Die Begründung ist die gleiche wie bei der **AL-Ersetzung**!

## PL1-Substitution (1)

Die **Substitution in PL1**  $\longrightarrow \varphi \mapsto \varphi[x_1, \dots, x_n / \tau_1, \dots, \tau_n]$

ersetzt keine Aussagen-, sondern **Objektvariablen**, nämlich:  
 paarweise verschiedene Objektvariablen  $x_1, \dots, x_n$   
 durch (nicht unbedingt verschiedene) Terme  $\tau_1, \dots, \tau_n$ .

**Termebene:**

$$\begin{array}{l}
 x \in OV \cup OK: \quad \overbrace{x[x_1, \dots, x_n / \tau_1, \dots, \tau_n]}^{sub} \quad := \begin{cases} \tau_j & \text{für } x = x_j \\ x & \text{sonst} \end{cases}, \\
 f \in FS \quad \quad f(\omega_1, \dots, \omega_k)_{[sub]} \quad := f(\omega_{1[sub]}, \dots, \omega_{k[sub]})
 \end{array}
 \left. \vphantom{\begin{array}{l} x \in OV \cup OK: \\ f \in FS \end{array}} \right\} \text{wie zu erwarten}$$

**Übergang zur Aussagenebene:**

$$R \in RS \quad R(\omega_1, \dots, \omega_k)_{[sub]} \quad := R(\omega_{1[sub]}, \dots, \omega_{k[sub]})$$

## PL2-Substitution (2) und gebundene Umbenennung

### Aussagenebene:

Negation	$(\neg\varphi)_{[sub]}$	$:=$	$\neg(\varphi_{[sub]})$	}	wie zu erwarten
$\otimes \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ :	$(\varphi \otimes \rho)_{[sub]}$	$:=$	$\varphi_{[sub]} \otimes \rho_{[sub]}$		

Quantoren	$(\forall x \varphi)_{[sub]}$	$:=$	$\forall x(\varphi_{[sub \neq x]})$	}	!
	$(\exists x \varphi)_{[sub]}$	$:=$	$\exists x(\varphi_{[sub \neq x]})$		

$sub_{\neq x}$  ist „wie  $sub$ , aber das gebundene  $x$  wird nicht ersetzt“.

Warum?

### Satz über die gebundene Umbenennung

Sei  $\varphi$  eine Formel, die die Variable  $y$  nicht frei enthält.

Dann ist  $\exists x \varphi \equiv \exists y \varphi_{[x/y]}$  und

$$\forall x \varphi \equiv \forall y \varphi_{[x/y]} .$$

## Nicht-Substituierbarkeit

Was im Allgemeinen gilt,  
gilt doch auch im Besonderen ...  
Also gilt doch für jede Substitution:  
 $\forall x \varphi \rightarrow \varphi_{[x/\tau]}$  ?

**NEIN!**

Wie kann das schief gehen?

z.B. so, mit  $[x/y]$ :

$$\forall x (\neg \forall y x = y) \rightarrow \neg \forall y y = y \quad ??$$

$$\forall x (\exists y x < y) \rightarrow \exists y y < y \quad ??$$

Aha, also wenn dadurch in  $\varphi$   
„ein freies Variablenvorkommen  
aus  $\tau$  **gebunden** wird!“

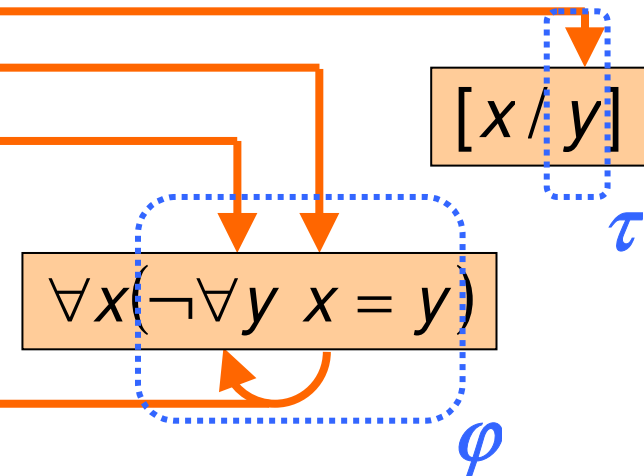
**Wie formalisieren wir das?**

## (Nicht-)Substituierbarkeit & Substitutionssatz

Term  $\tau$  ist **nicht erlaubt** (nicht frei, nicht substituierbar) für  $x$  in  $\varphi$ ,  
 $\neg \text{free}(\tau, x, \varphi)$ ,

$:\Leftrightarrow$  Es existieren ...

- Variablenvorkommen  $[y]$  in  $\tau$ ,
- ein freies Variablenvorkommen  $[x]$  in  $\varphi$ ,  $x \neq y$
- ein Quantorenvorkommen  $[Qy]$  in  $\varphi$ , so dass
- dieses  $[x]$  im Scope von  $[Qy]$  liegt



**Substitutionssatz**  
 $\text{free}(\tau, x, \varphi) \Rightarrow \forall x \varphi \rightarrow \varphi_{[x/\tau]}$

## Substituierbar (d.h. $\tau$ frei für $x_i$ in $\varphi$ )? – ein Algorithmus

Geg.:  $\tau$ ,  $x_i$  und  $\varphi$  ( $i$  ist fest)

1) Bestimme alle freien  $[x_i]$  (freie Vorkommen des gegebenen  $x_i$ ) in  $\varphi$ .

Gibt es keine  $\rightarrow \tau$  ist **frei** für  $x_i$  in  $\varphi$  & Stop.

2) Für alle freien  $[x_i]$  in  $\varphi$ :

Bestimme in  $\varphi$  alle Quantisierungen  $[Qx_k]$ , automatisch mit  $i \neq k$ , in deren Scope dieses  $[x_i]$  liegt.

(i) Gibt es keine  $\rightarrow \tau$  ist **frei** für  $x_i$  in  $\varphi$  & Stop.

(ii) Für alle diese  $[Qx_k]$ :

Kommt  $x_k$  in  $\tau$  vor  $\rightarrow \tau$  ist **nicht frei** für  $x_i$  in  $\varphi$  & Stop.

3) [ Sind nun also alle freien Vorkommen  $[x_i]$  in  $\varphi$  durchprobiert, ohne auf Stop zu laufen  $\rightarrow$  ]

$\tau$  ist **frei** für  $x_i$  in  $\varphi$  & Stop.



Manche sagen auch „ $x$  frei für  $\tau$ “ anstatt „ $\tau$  frei für  $x$ “.



## Einfache Folgerungen

- $\tau$  enthält keine Variable  $\rightarrow$   $\tau$  frei für alle Variablen  
in allen Formeln.
- keine Variable in  $\tau$  gebunden in  $\varphi \rightarrow \tau$  frei für alle Variablen in  $\varphi$ .
- stets:  $x_i$  ist frei für  $x_i$  in (jedem)  $\varphi$ .
- $\varphi$  enthält kein freies  $[x_i]$   $\rightarrow$  (jedes)  $\tau$  ist frei für  $x_i$  in  $\varphi$ .

# Prädikatenlogik

- Syntax
- Semantik
- Formeln, Modelle, Tautologien und Anwendungen
- **Entscheidbarkeitsfragen**

**PL1-Vorteil:** Fast die gesamte Mathematik lässt sich damit ausdrücken.

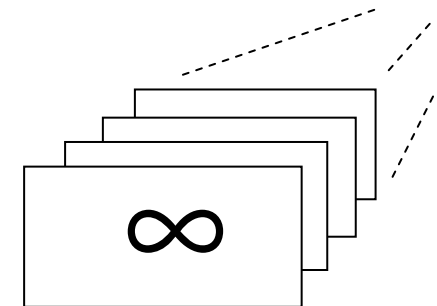
In der **Aussagenlogik** sind **semantische Eigenschaften** – bis evtl. auf Folgerung aus unendlichen Formelmengen – prinzipiell **entscheidbar**: die ...

Wahrheitstafel

... erlaubt die vollständige Prüfung anhand aller (endlich vielen!) **Interpretationen** (= AL-Belegungen).

**PL1-Nachteil**

Jetzt, in der Prädikatenlogik, wären **unendlich viele Strukturen** und pro unendlicher Struktur jeweils **unendlich viele Variablen-Interpretationen** zu untersuchen.



## (Un-)Entscheidbarkeit (1)

Es gibt keinen Algorithmus, der

- Erfüllbarkeit,
- Allgemeingültigkeit oder
- Folgerung

**entscheidet.**

aber: **Beweise**  
können algorithmisch **überprüft**  
werden!

**Nie** – oder nur **noch keinen?**

... weil lediglich **noch keiner schlaug genug** war einen zu (er)finden?

**Nein:**

„Vermutlich endgültig“,  
weil es „**fast** beweisbar“  
**keinen geben kann!** (Church, Turing 1936)

**Wie zeigt man so etwas?**  
**Und wieso „fast“?**

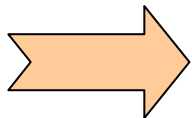
## (Un-)Entscheidbarkeit (2): Churchsche These

**Mathematische Definition** des Begriffs des (beliebigen!) **Algorithmus**, z.B.

- **Turing-Maschine** (TM): elementare Programmschritte + Speicher
- andere formalisierte Algorithmusbegriffe  
(Rekursive Funktionen, Markov-Systeme, Post-Systeme, ...)

Erfahrung (bewiesen!):

Alle bisherigen Algorithmus-Klassen berechnen die gleiche Klasse von Funktionen.



**Churchsche These:**  
Die Klasse der intuitiv berechenbaren Funktionen ist genau die Klasse der Turing-berechenbaren Funktionen.

Was also mit Turing-Maschinen nicht berechenbar ist, dafür gibt es – laut Churchscher These – überhaupt keinen Algorithmus.

Also reden wir jetzt mal nur noch über Turing-Maschinen.

## (Un-)Entscheidbarkeit (3): Halteproblem

### Technische Überlegungen →

Wenn eine TM über Erfüllbarkeit von PL1-Formeln entscheiden könne, ließe sie sich umformen zu einer einer TM, die entscheidet, ob eine gegebene TM auf einem gegebenen Input mit einem Ergebnis stoppt.

### Halteproblem

Es gibt aber keine TM, die für jede TM und beliebige Eingaben entscheidet, ob diese TM auf dieser Eingabe terminiert!

### Beweisskizze

**Annahme**, es ginge **doch**, sagen wir mit der TM  $T_{super}$ .

Zähle **alle** Turingmaschinen durch:  $T_1, T_2, T_3, T_4, \dots$  (**geht**: Gödel-Nummern!).

Definiere neue TM  $T_{seltsam}$  durch

$$T_{seltsam}(T_i) := \begin{cases} T_i(T_i) + 1 & \text{wenn } T_i(T_i) \text{ existiert} \\ 0 & \text{wenn } T_i(T_i) \text{ nicht stoppt} \end{cases}$$

Wegen  
 $T_{super}$   
berechenbar!!

**Diagonalargument:** Dann kann  $T_{seltsam}$  nicht eine der  $T_i$  sein.

→ ein **Widerspruch**; die **Annahme** muss **falsch** sein.

## Halbentscheidungs-Algorithmen

Eine Frage(nmenge) ist durch Algorithmus *Alg* **halbentscheidbar**  $:\Leftrightarrow$

- Falls korrekte Antwort JA  $\Rightarrow$  *Alg* endet irgendwann mit JA;
- Falls NEIN: *Alg* endet nicht

wieso?

Für die **Allgemeingültigkeit** (also auch **Unerfüllbarkeit** und **Folgerung**) lernen wir noch

**Beweisregeln** und **Halbentscheidungs-Algorithmen** kennen:

- Beweiskalküle
- Beweisalgorithmus
- PL1-Tableaux
- PL1-Resolution
- Natürliches Schließen (Werkzeugkasten)

wieso?

**Also** kann es **keinen Halbentscheidungs-Algorithmus** für die **Erfüllbarkeit** (**Widerlegbarkeit**, **Nicht-Folgerung**) geben!

Und wenn ich das Gegenteil halb-entscheidbar auf Unerfüllbarkeit prüfe??

# Anwendung pfiffiger Beweismethoden

## (1) Diagonalverfahren

Was haben

- der Beweisgedanke der **Unentscheidbarkeit des Halteproblems** und
  - der Standardbeweis für die **Überabzählbarkeit der reellen Zahlen**
- gemeinsam?

Zu einer unendlichen Matrix mit Zeilen  $z_i$  wird per geschickter\* **Änderung aller Diagonalelemente** eine „neue Zeile“ generiert, die sich von der  $i$ -ten Matrixzeile  $z_i$  an der  $i$ -ten Stelle unterscheidet und daher **nicht** in der Aufzählung  $z_1, z_2, z_3, \dots$  **vorkommen kann**.

$z_1$ :	$a_{11}$	$a_{12}$	$a_{13}$	$\dots$
$z_2$ :	$a_{21}$	$a_{22}$	$a_{23}$	$\dots$
$z_3$ :	$a_{31}$	$a_{32}$	$a_{33}$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

Beim **Halteproblem** ist  $a_{ik}$  das Ergebnis der Anwendung von Maschine  $T_i$  auf das Programm der Maschine  $T_k$  (ggf. *undefiniert* wegen Nichtterminierung – die angenommene  $T_{super}$  sagt uns wann das der Fall ist!). Bei den **reellen Zahlen** (sagen wir: zwischen 0 und 1) ist  $a_{ik}$  die  $k$ -te Dezimalstelle hinter dem Komma.

\*) In beiden Fällen muss eine **legitime Zeile** erzeugt werden!



## Anwendung pfiffiger Beweismethoden

### (2) Koordinierte Halbentscheidungsalgorithmen

Wieso kann es keinen Halbentscheidungsalgorithmus für die Erfüllbarkeit von PL1-Formeln geben?

- **Angenommen es gäbe einen: ERF.**
- GRES (Grundresolution) ist ein Halbentscheidungsalgorithmus für Unerfüllbarkeit – den lernen wir noch kennen.
- Nun programmiert man leicht einen übergeordneten Algorithmus ÜA, der abwechselnd einen „Schritt“ von ERF und einen von GRES durchführt und wendet ihn jeweils auf eine (evtl. geeignet umgeformte) PL1-Formel  $\varphi$  an. ÜA soll stoppen sobald er mit ERF oder GRES auf ein *Stop* läuft.
- $\varphi$  ist entw. erfüllbar oder unerfüllbar, d.h. ERF oder GRES läuft in endlich vielen Schritten auf ein (bejahendes!) *Stop*, sagen wir: nach  $n$  Schritten.
- D.h. ÜA wäre ein Algorithmus, der (bei  $\varphi$  in max.  $2n$  Schritten) entscheidet ob  $\varphi$  erfüllbar ist. Damit wäre das Halteproblem lösbar – Widerspruch!
- Also gibt es dieses ÜA nicht, also muss die Annahme falsch sein.
- **Dieses ERF gibt es nicht. (Widerspruchsbeweis)**

# Prädikatenlogik

- Syntax
- Semantik
- Formeln, Modelle, Tautologien und Anwendungen
- Entscheidbarkeitsfragen
- **Kalküle und Tableaux**

# Ein(e Gruppe von) Prädikatenkalkül(en) (1)

## PL1-Formeln

- Bekannt (mit Aussagevariablen)

## Regeln

- Modus Ponens

$$\text{MP} = \frac{\varphi, \varphi \rightarrow \psi}{\psi}$$

## Ein(e Gruppe von) Prädikatenkalkül(en) (2)

**Axiome(nschemata)** (für bel. Formel  $\varphi$  und bel. Variable  $x$ )

- ein AL-Axiomensystem

Aussagenlogik AL

und beliebige Verallgemeinerungen  $\forall x_1 \dots \forall x_n$  von Formeln der Arten ...

- $\varphi[A_1, \dots, A_k / \varphi_1, \dots, \varphi_k]$  wobei  $\varphi$  AL-Tautologie,  $\varphi_i$  Formeln (AL-Gültigkeit AL-G)
- $(\forall x\varphi) \rightarrow \varphi[x/\tau]$  sofern  $\tau$  Term, der für  $x$  in  $\varphi$  erlaubt ist:  $\text{free}(\tau, x, \varphi)$  (Substitutionssatz, Spezialisierung SP)
- $\varphi \rightarrow (\forall x\varphi)$  sofern  $x$  nicht frei in  $\varphi$  vorkommt (Generalisierung GN)
- $\forall x(\varphi \rightarrow \psi) \rightarrow (\forall x\varphi \rightarrow \forall x\psi)$  (Generalisierungsdistribution GD)
- $\exists x\varphi \leftrightarrow \neg \forall x \neg \varphi$  (Existenz„definition“ ED)

## Anwendung des Prädikatenkalküls (1)

### Korrektheit dieser PL1-Kalküle

Alle ableitbaren Formeln sind allgemeingültig.

Alle aus einer Formelmenge  $M$  ableitbaren Formeln sind Folgerungen von  $M$ .

### Gödelscher Vollständigkeitssatz:

... und umgekehrt! D.h. alle allgemeingültigen Formeln und Folgerungen sind damit ableitbar.

warum  
fordern?

### Generalisierungssatz

$M \vdash \varphi$  und  $x$  in keinem  $\psi \in M$  frei  $\Rightarrow M \vdash \forall x \varphi$

**GS**

### Deduktionssatz:

$M \cup \{\varphi\} \vdash \psi \Rightarrow M \vdash \varphi \rightarrow \psi$

**DS**

## Anwendung des Prädikatenkalküls (2)

### Beispiel für eine Meta-Ableitung (d.h. unter Verwendung semantischer Sätze)

- |    |   |                         |
|----|---|-------------------------|
| 1. | $(A_1 \rightarrow \neg A_2) \rightarrow (A_2 \rightarrow \neg A_1)$   | aus AL, MP (mehrfach)   |
| 2. | $(\forall y \neg P(y) \rightarrow \neg P(x)) \rightarrow (P(x) \rightarrow \neg \forall y \neg P(y))$                     | 1, AL-G                 |
| 3. | $\forall x [(\forall y \neg P(y) \rightarrow \neg P(x)) \rightarrow (P(x) \rightarrow \neg \forall y \neg P(y))]$         | 2, GS ( $M=\emptyset$ ) |
| 4. | $\forall x (\forall y \neg P(y) \rightarrow \neg P(x)) \rightarrow \forall x (P(x) \rightarrow \neg \forall y \neg P(y))$ | 3, GD                   |
| 5. | $\forall y \neg P(y) \rightarrow \neg P(x)$   | SP                      |
| 6. | $\forall x (\forall y \neg P(y) \rightarrow \neg P(x))$   | 5, GS                   |
| 7. | $\forall x (P(x) \rightarrow \neg \forall y \neg P(y))$   | 4, 6, MP                |

**Meta-Ableitungen garantieren die Existenz von  
– meist deutlich längeren – Ableitungen  
(bzw. liefern sie).**

## Halbentscheidung mit „roher Gewalt“

Wir wollen  $\varphi$  auf **Allgemeingültigkeit** prüfen.

Dazu **produzieren** wir mittels eines PL1-Kalküls **systematisch\*** der Reihe nach **alle möglichen Ableitungen**.

Nach jeder solchen Produktion:

Erzeugte **Ableitung endet mit  $\varphi$ ?**

ja: JA, Stop.

nein: nächste Produktion.

Man kann aber auch gezielter vorgehen:

PL1-Algorithmen (Tableaux, Resolution etc.)

\*) **Wie geht das?**  
**(So nicht:**

SP hat doch  $\infty$  viele Varianten!

Stufe  $n$ :  $n$  Regel-Anwendungen auf alle Axiome.  $n=0, 1, \dots$ )

## PL1-Tableaux (1)

Anwendung für **geschlossene Formeln** (ohne freie Variablen)

**Zusätzliche Regeln (zu den aussagenlogischen Tableauregeln)**

**Speziali-  
sierung**  $Sp_+ = \frac{\forall x\varphi}{\varphi[x/\tau]}$

$$Sp_- = \frac{\neg\exists x\varphi}{\neg\varphi[x/\tau]}$$

$\tau$  für  $x$  in  $\varphi$  zulässig,  
d.h.  $\text{free}(\tau, x, \varphi)$

**Witness  
(Zeuge)**  $W_+ = \frac{\exists x\varphi}{\varphi[x/c]}$



$$W_- = \frac{\neg\forall x\varphi}{\neg\varphi[x/c]}$$

$c$  Konstanten-  
symbol, das im  
Zweig neu ist



## PL1-Tableaux – Anwendungsbeispiel

Zu beweisen:  $\forall x(P(x) \vee Q(x)) \rightarrow (\exists xP(x) \vee \forall xQ(x))$

Gegenbeh.:	1.	$\neg[\forall x(P(x) \vee Q(x)) \rightarrow (\exists xP(x) \vee \forall xQ(x))]$	Regel / Prämisse
	2.	$\forall x(P(x) \vee Q(x))$	(TR1) 1
	3.	$\neg(\exists xP(x) \vee \forall xQ(x))$	(TR1) 1
	4.	$\neg\exists xP(x)$	(TR3) 3
	5.	$\neg\forall xQ(x)$	(TR3) 3
	6.	$\neg Q(c)$ 	W_ 5
	7.	$\neg P(c)$ 	Sp_ 4
	8.	$P(c) \vee Q(c)$	Sp+ 2
	9.	<u><math>P(c)</math></u>	(TR8) 8
	10.	<u><math>Q(c)</math></u>	

---

Tableau- regeln AL, Erinnerung	1.	$\frac{\neg(\varphi \rightarrow \psi)}{\varphi \quad \neg\psi}$	3.	$\frac{\neg(\varphi \vee \psi)}{\neg\varphi \quad \neg\psi}$	8.	$\frac{\varphi \vee \psi}{\varphi \mid \psi}$
--------------------------------------	----	---	----	--	----	---

## PL1-Tableaux-Eigenschaften

Als **Kalkül** ist PL1-Tableaux **korrekt** und **vollständig**:

- Enthalten alle Zweige einen Widerspruch zwischen zwei Literalen, ist die Wurzel unerfüllbar bzw. ihre Negation allgemeingültig
- Ist die Wurzel unerfüllbar bzw. ihre Negation allgemeingültig, so existieren abgeleitete Tableaux, die in allen Zweigen einen Widerspruch zwischen zwei Literalen enthalten.

Als **Algorithmus** ist PL1-Tableaux leider **nicht** unbedingt **terminierend**.

Bei geeigneter Variation und Steuerung, z.B.

- im Rahmen eines Brute-Force-Verfahrens oder
- (weitaus effizienter) mit raffinierteren Techniken

taugen sie als **Halbentscheidungsverfahren**:

Die Terminierung bei unerfüllbarer Wurzel kann garantiert werden.
---

## PL1-Werkzeug-Ergänzungs-Kasten (1)

All-Benutzung Spezialisierung	Sp $\frac{\forall x\varphi}{\varphi_{[x/\tau]}}$	( $\tau$ varia- blenfrei)	Existenz- Einführung	EE $\frac{\varphi_{[x/\tau]}}{\exists x\varphi}$	( $\tau$ varia- blenfrei)
Existenz- Benutzung	EB $\frac{\exists x\varphi}{\varphi_{[x/c]}}$	(c neu, vgl. All-Beweis) <sup>3</sup>			
Nicht-Existenz- Benutzung	NEB $\frac{\neg\exists x\varphi}{\forall x\neg\varphi}$		Nicht-All- Benutzung	NAB $\frac{\neg\forall x\varphi}{\exists x\neg\varphi}$	

**All-Beweis-Schema AB:** Zeige  $\forall x\varphi$ , unmittelbar gefolgt von weiterem Blockbeginn: Zeige  $\varphi_{[x/c]}$  (c im Beweis **neuer Konstantenname**) + Blockbeginn. Sobald im gleichen Block  $\varphi_{[x/c]}$  erreicht: Blockende. Dann weiteres Blockende mit  $\forall x\varphi$  (AB).

Zeige $\forall x\varphi$	
Zeige $\varphi_{[x/c]}$	
:	
$\varphi_{[x/c]}$	(...B)
$\forall x\varphi$	(AB)

<sup>3</sup> Kein Beweis der Formel mit c, nur eine temporäre Namensgebung: „Nennen wir es c!“

## PL1-Werkzeug-Ergänzungs-Kasten (2)

Der AL&PL1-Werkzeugkasten  
ist **korrekt** und **vollständig**  
für **geschlossene** PL1-Formeln.

### 2 Beweisstrategien für Existenzaussagen:

#### Direkter Beweis:

Zeige $\exists x\varphi$	
$\vdots$	
$\varphi_{[x/\tau]}$	
$\exists x\varphi$	(EE)
$\exists x\varphi$	(DB)

#### Indirekter Beweis:

Zeige $\exists x\varphi$	
$\neg\exists x\varphi$	(Ann)
$\vdots$	
$\perp$	(WE)
$\exists x\varphi$	(IB)

# Beispiel 1 für die Arbeit mit dem kompletten Werkzeugkasten

Beweisen Sie die Korrektheit von

$$\forall xP(x) \rightarrow \exists yP(y)$$

Zeige  $\forall xP(x) \rightarrow \exists yP(y)$

(1)		$\forall xP(x)$	Ann.
		Zeige $\exists yP(y)$	
(2)		$P(a)$	1, Sp
(3)		$\exists yP(y)$	2, EE
(4)		$\exists yP(y)$	DB
(5)		$\forall xP(x) \rightarrow \exists yP(y)$	BB

## Beispiel 2 für die Arbeit mit dem kompletten Werkzeugkasten

**Beweisen Sie die Korrektheit der Folgerung**

$$\{\forall x(P(x) \rightarrow Q(x)), \neg Q(a)\} \models \neg \forall y P(y)$$

(1)	$\forall x(P(x) \rightarrow Q(x))$	Geg.
(2)	$\neg Q(a)$	Geg.
	Zeige $\neg \forall y P(y)$	
(3)	$\forall y P(y)$	Ann.
(4)	$P(a)$	3, Sp
(5)	$P(a) \rightarrow Q(a)$	1, Sp
(6)	$Q(a)$	4, 5, MP
(7)	$\perp$	2, 6, WE
(8)	$\neg \forall y P(y)$	IB

# Beispiel 3 für die Arbeit mit dem kompletten Werkzeugkasten

**Beweisen Sie**

$$\forall x(\neg P(x) \rightarrow \neg \forall y P(y))$$

	Zeige $\forall x(\neg P(x) \rightarrow \neg \forall y P(y))$	
	Zeige $\neg P(c) \rightarrow \neg \forall y P(y)$	
(1)	$\neg P(c)$	Ann.
	Zeige $\neg \forall y P(y)$	
(2)	$\forall y P(y)$	Ann.
(3)	$P(c)$	2, Sp
(4)	$\perp$	1,3, WE
(5)	$\neg \forall y P(y)$	IB
(6)	$\neg P(c) \rightarrow \neg \forall y P(y)$	BB
(7)	$\forall x(\neg P(x) \rightarrow \neg \forall y P(y))$	AB