Prädikatenlogik

Von der Aussagenlogik zur Prädikatenlogik

- Prädikatenlogik ist eine Erweiterung der Aussagenlogik
  - Syntax der Prädikatenlogik
  - Semantik der Prädikatenlogik

Was unterscheidet die Prädikatenlogik von der Aussagenlogik?

- Größere Ausdrucksstärke: feinere Differenzierungen

➔ Größerer Aufwand bei Berechnung semantischer Eigenschaften und Beziehungen

Von der Aussagenlogik zur Prädikatenlogik: Motivation

Aussagenlogik

- Aussagen sind die kleinsten bedeutungstragenden Einheiten

Beispiele

- Hamburg ist eine Stadt.
- München ist eine Stadt.
- Hamburg liegt nördlich von München.
- München liegt südlich von Hamburg.

Atomare Formeln AL Aussagensymbole	Atomare Formeln PL Interne Struktur
Shh	S1(hh)
Sm	S1(m)
Nhh_m	N(hh, m)
Sm_hh	S2(m, hh)

Prädikatenlogik

- berücksichtigt die interne Struktur von Aussagen und erlaubt es, zusätzlich bestimmte Beziehungen zwischen ‘Objekten’ zum Ausdruck zu bringen.

➔ Sprache mit größerer Ausdruckskraft.

- Konsequenzen (z.B. im Datenbankbereich):
  Systematische Beziehungen zwischen Aussagen und Fragen:
  - Welche Städte liegen südlich von Hamburg und nördlich von München?
  - Frankfurt liegt südlich von Hamburg und nördlich von München.

Die kleinsten Einheiten der Prädikatenlogik

… haben verschiedene syntaktische Kategorien

- *Terme* repräsentieren Objekte.
- *Prädikatensymbole* (Relationssymbole) repräsentieren Eigenschaften und Relationen.
- *Formeln* repräsentieren Aussagen.

PL-Repräsentation	Syntaktische Kategorie	Übersetzungsschlüssel	Syntaktische Kategorie
hh	Konstante, Term	Hamburg	Name
m	Konstante, Term	München	Name
S1	Prädikatensymbol	ist eine Stadt	Prädikat
N	Prädikatensymbol	liegt nördlich von	
S2	Prädikatensymbol	liegt südlich von	

PL-Repräsentation	Syntaktische Kategorie	Übersetzung	
S1(hh)	(atomare) Formel	Hamburg ist eine Stadt.	Satz
S1(m)	(atomare) Formel	München ist eine Stadt.	Satz
N(hh, m)	(atomare) Formel	Hamburg liegt nördlich von München.	Satz
S2(m, hh)	(atomare) Formel	München liegt südlich von Hamburg.	Satz

Terme: Eindeutige Objektbenennungen (Namen)

Konstanten: atomare Terme

PL-Repräsentation	syntaktische Kategorie	Übersetzung
hh	Konstante, (atomarer) Term	Hamburg
m	Konstante, (atomarer) Term	München
d	Konstante, (atomarer) Term	Deutschland

Funktionssymbole

hs	einstelliges Funktionssymbol	die Hauptstadt von
ewz	einstelliges Funktionssymbol	die Einwohnerzahl von
abst	zweistelliges Funktionssymbol	der Abstand zwischen

Komplexe Terme: Funktionssymbole, kombiniert mit der richtigen Anzahl von Termen

hs(d)	(komplexer) Term	die Hauptstadt von Deutschland
abst(hh, m)	(komplexer) Term	der Abstand zwischen Hamburg und München
ewz(hh)	(komplexer) Term	die Einwohnerzahl von Hamburg
ewz(hs(d))	(komplexer) Term	die Einwohnerzahl der Hauptstadt von Deutschland

Prädikatensymbole (Relationssymbole)

- kombiniert mit der richtigen Anzahl von Termen ergeben (atomare) Formeln.

PL-Repräsentation	syntaktische Kategorie	Übersetzung
S1	einstelliges Prädikatensymbol	ist eine Stadt
S1(hh)	(atomare) Formel	Hamburg ist eine Stadt.
N	zweistelliges Prädikatensymbol	liegt nördlich von
N(hh, m)	(atomare) Formel	Hamburg liegt nördlich von München.
Gr	zweistelliges Prädikatensymbol	ist größer als
Gr(hh, m)	(atomare) Formel	Hamburg ist größer als München.

- Komplexe Terme können in Formeln auftreten

Gr(ewz(hh), ewz(m))	Die Einwohnerzahl von Hamburg ist größer als die Einwohnerzahl von München.
Gr(ewz(hs(d)), ewz(hh))	Die Hauptstadt von Deutschland hat mehr Einw. als HH.
¬Gr(abst(hh, m), abst(m, hh))	Der Abstand zwischen HH und München ist nicht größer als der Abstand zwischen München und HH.
Gr(abst(hh, m), abst(m, hs(d)))	Der Abstand zwischen Hamburg und München ist größer als der Abstand zwischen München und der Hauptstadt von Deutschland.

Prädikatensymbole – Relationale Datenbanksysteme

PL-Formel:

Flugverb(ab8862, ham, zrh, 240507, 06:30, 07:50)

Übersetzung:

Flug 'AB8862' fliegt am 24.05.07 von Hamburg nach Zürich; Abflug ist um 6:30, Ankunft um 7:50.

Anmerkungen:

- Flugverb ist ein 6-stelliges Prädikaten- / Relationensymbol
- ab8862, ham, zrh, 240507, 06:30, 07:50 sind Konstanten

Relationale Datenbank

- Variablenfreie PL-Formeln zu einem Prädikatensymbol werden in Tabellen zusammengeführt, z.B

Flugverb	Flight-No.	Date	Origin	Time Depart.	Destination	Time Arriv.
	ab8862	240507	ham	06:30	zrh	07:50
	ab8780	240507	ham	15:15	zrh	16:35
	ab8862	250507	ham	06:30	zrh	07:50

Quantoren und Variablen

- erlauben Aussagen über Objekte ohne eindeutige Bezugnahme

$\exists x\ (S1(x) \land N(x, m))$	Es gibt eine Stadt, die nördlich von München liegt.
$\neg\exists x\ (S1(x) \land S2(x, x))$	Keine Stadt liegt südlich von sich selbst.
$\exists x\ (S1(x) \land Gr(ewz(x), ewz(hh)))$	Eine Stadt hat mehr Einwohner als Hamburg.
$\neg\exists x\ (S1(x) \land Gr(ewz(x), ewz(hs(d))))$	Keine Stadt hat mehr Einwohner als die Hauptstadt von Deutschland.
$\forall x\ \forall y\ ((S1(x) \land (S1(y) \land N(y, x))) \Rightarrow S2(x, y))$	Jede Stadt liegt südlich von jeder Stadt, die nördlich von ihr liegt.
$\forall y\ \forall x\ ((S1(y) \land (S1(x) \land N(y, x))) \Rightarrow S2(x, y))$	Wenn eine Stadt nördlich von einer zweiten Stadt liegt, dann liegt die zweite Stadt südlich von der ersten.
$(S1(hh) \land N(hh, m)) \Rightarrow \exists x\ (S1(x) \land N(x, m))$	Wenn Hamburg eine Stadt ist und nördlich von München liegt, dann gibt es eine Stadt, die nördlich von München liegt.

Prädikatensymbole – Datenbanksysteme (Fortsetzung)

Variablen

- spielen eine zentrale Rolle in der Formulierung von Anfragen
  - Flugverb(x, ham, zrh, 240507, y, z)
    kann als Aufforderung verstanden werden, ein Modell der Formel zu finden, unter geeigneter Belegung von x, y, und z.
  - Diese Sichtweise wird verwendet
    - in der Logischen Programmierung (Prolog) → Modul SE-3
    - bei Datenbankanfragen (SQL: Structured Query Language) → Modul GDB

Quantoren

- werden in Standard-Datenbankensystemen in der Regel nicht explizit verwendet
  - sie sind implizit in die Verfahren der Antwortfindung integriert
- spielen in Deduktiven Datenbanken & Wissensbasierten Systemen eine Rolle, z.B. um Regularitäten der Domäne zu modellieren:
  Der Übergang von Flug_1 zu Flug_2 klappt im Flughafen X, wenn die Abflugzeit mehr als 90 Minuten nach der Ankunftszeit liegt. (Formulierung in PL zur Übung.)

Syntax der Prädikatenlogik: Das Alphabet

Definition 9.1

Das Alphabet der Prädikatenlogik besteht aus

- einer abzählbaren Menge $\mathcal{V}_{PL}$, dem Inventar verfügbarer Symbole ($\mathcal{V} \approx$ Vokabular).
  Diese unterteilt sich in
  - einer Menge von Variablen: x, y, x_i mit $i = 1, 2, \ldots$
  - einer Menge von Konstanten: $a, b, c, a_i, \ldots$
  - einer Menge von Funktionssymbolen: $f, g, h, f_i, \ldots$
  - einer Menge von Prädikatensymbolen (Relationssymbolen): $P, Q, R, P_i, \ldots$
  - einer Menge von Aussagensymbolen: $A, A_i, \ldots$
- den logischen Symbolen
  - Junktoren: $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$
  - Quantoren: $\forall, \exists$
- den Hilfssymbolen Klammern und Komma:), (, ,
- Jedem Funktions- und jedem Prädikatensymbol ist eindeutig eine Stelligkeit $k > 0$ zugewiesen.

Zum Selbststudium

- Konstanten entsprechen Funktionssymbolen der Stelligkeit 0.
- Aussagensymbole entsprechen Prädikatensymbolen der Stelligkeit 0. Es ist nicht nötig, Aussagensymbole in der Prädikatenlogik zu verwenden.
- Wird dasselbe Symbol / Wort mit unterschiedlicher syntaktischen Kategorie (Stelligkeit) verwendet, dann verhält sich die Logik blind gegenüber der oberflächlichen Übereinstimmung und behandelt die unterschiedlichen Verwendungen unabhängig voneinander.

Bezeichnungskonventionen

- Als verfügbare Symbole werden oft auch Zeichenketten (Wörter) verwendet, insbesondere, wenn eine Assoziation mit einer bestimmten Bedeutung beabsichtigt ist.
- Wir verwenden bestimmte Buchstaben als (Meta-)Variablen/Platzhalter, deren Wert folgendes sein kann

PL-Variablen	u, v, w, x, y, z	Kleinbuchstaben (Ende des Alphabets)
Konstanten	a, b, c	Kleinbuchstaben (Anfang des Alphabets)
Funktionssymbole	f, g, h	Kleinbuchstaben (Bereich f – h)
Prädikatensymbole	P, Q, R	Großbuchstaben (Bereich P – S)
Formeln	F, G, H	Großbuchstaben (Bereich F – H)
Terme	t, r, s	Kleinbuchstaben (Bereich r – t)

Syntax der Prädikatenlogik: Terme von $\mathcal{L}_{PL}$

Definition 9.2 (Terme der Prädikatenlogik)
Gegeben sei ein Inventar verfügbarer Symbole $\mathcal{V}_{PL}$.
Die Menge der prädikatenlogischen Terme ist wie folgt definiert:

T-1 Jede Variable ist ein Term.
T-2 Jede Konstante ist ein Term.
T-3 Falls f ein Funktionssymbol der Stelligkeit k ist, und falls $t_1, \ldots, t_k$ Terme sind, so ist $f(t_1, \ldots, t_k)$ ein Term.
T-4 Es gibt keine anderen Terme, als die, die durch endliche Anwendung der Schritte T-1 bis T-3 erzeugt werden.

- Der Term t ist Teilterm des Terms t', wenn t beim Aufbau von t' (Schritt T-3) verwendet wurde.

Beispiel: ewz(hs(d)) ≈ „Einwohnerzahl der Hauptstadt von Deutschland“

Bemerkung zu Def. 9.2

- Entsprechend der Definition lassen sich
  - induktive Beweise über die Struktur der Terme führen
  - Funktionen rekursiv über die Struktur der Terme definieren.

Zum Selbststudium: Prinzip der strukturellen Induktion für Terme der Prädikatenlogik

Um zu beweisen, dass eine Behauptung $\mathcal{B}(t)$ für jeden Term t der Prädikatenlogik gilt, genügt es, folgende Schritte durchzuführen:

Induktionsanfang: Man zeigt, dass $\mathcal{B}(a)$ für jede Konstante a gilt, und dass $\mathcal{B}(x)$ für jede Variable x gilt.

Induktionsannahme: Man nimmt an, dass f ein beliebiges Funktionssymbol ist, benennt die Stelligkeit von f als k und nimmt an, dass $t_1, \ldots, t_k$ Terme sind, die alle die Behauptung erfüllen, also, dass $\mathcal{B}(t_1), \ldots, \mathcal{B}(t_k)$ gelten.

Induktionsschritt: Man zeigt, dass dann auch $\mathcal{B}(f(t_1, \ldots, t_k))$ gilt.

Syntax der Prädikatenlogik: Formeln von $\mathcal{L}_{PL}$

Definition 9.3 (Formeln der Prädikatenlogik)
Gegeben sei ein Inventar verfügbarer Symbole $\mathcal{V}_{PL}$.
Die Menge der prädikatenlogischen Formeln ($\mathcal{L}_{PL}$) ist wie folgt definiert:

F-1 Jedes Aussagensymbol ist eine (atomare) Formel.

F-2 Falls P ein Prädikatensymbol der Stelligkeit k ist, und falls $t_1, \ldots, t_k$ prädikatenlogische Terme sind, so ist $P(t_1, \ldots, t_k)$ eine (atomare) Formel.

F-3 Falls F und G Formeln sind, so sind auch folgende Zeichenketten (komplexe) Formeln: $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \Rightarrow G)$, $(F \Leftrightarrow G)$

F-4 Ist x eine Variable und F eine Formel, so sind $\exists x\ F$ und $\forall x\ F$ (komplexe) Formeln.

F-5 Es gibt keine anderen Formeln, als die, die durch endliche Anwendung der Schritte F-1 bis F-4 erzeugt werden.

- Die Formel F ist Teilformel der Formel H, wenn F beim Aufbau von H (Schritt F-3, F-4) verwendet wurde.
- Der Term t ist Teilterm der Formel H, wenn t beim Aufbau von H (Schritt F-2) verwendet wurde oder Teilterm eines Terms ist, für den dies gilt.
- Bezüglich der Formeln $\exists x\ F$ und $\forall x\ F$ bezeichnen wir x als die Quantorenvariable.

Zum Selbststudium: Prinzip der strukturellen Induktion für prädikatenlogische Formeln

Bemerkung zu Def. 9.3

- Entsprechend der Definition lassen sich
  - induktive Beweise über die Struktur der prädikatenlogischen Formeln führen
  - Funktionen rekursiv über die Struktur der prädikatenlogischen Formeln definieren.

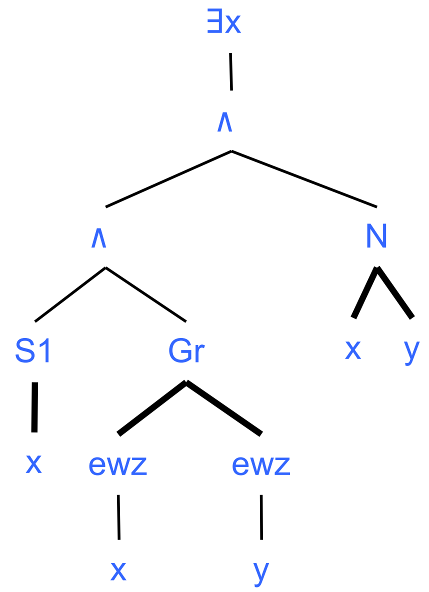
Um zu beweisen, dass eine Behauptung $\mathcal{B}(F)$ für jede Formel $F \in \mathcal{L}_{PL}$ gilt, genügt es, folgende Schritte durchzuführen:

Induktionsanfang: Man zeigt, dass $\mathcal{B}(F)$ für jede atomare Formel F gilt, also für die Aussagensymbole und für alle Formeln der Form $P(t_1, \ldots, t_k)$, wobei P ein k-stelliges Prädikatensymbol ist und $t_1, \ldots, t_k$ Terme sind.

Induktionsannahme: Man nimmt an, dass F und G Formeln sind, für die $\mathcal{B}(F)$ und $\mathcal{B}(G)$ gelten, und dass x eine Variable ist.

Induktionsschritt: Man zeigt, dass dann auch $\mathcal{B}(\neg F)$, $\mathcal{B}((F \wedge G))$, $\mathcal{B}((F \vee G))$, $\mathcal{B}((F \Rightarrow G))$, $\mathcal{B}((F \Leftrightarrow G))$, $\mathcal{B}(\exists x\ F)$ und $\mathcal{B}(\forall x\ F)$ gelten.

Prädikatenlogische Strukturbäume

	$\exists x\ ((S1(x) \land Gr(ewz(x), ewz(hh))) \land N(x, y))$
Quantoren bilden innere Knoten mit einem Nachfolger. Diese Knoten sind auch mit der zugehörigen Quantorenvariable gekennzeichnet. Prädikatensymbole und Funktionssymbole bilden innere Knoten. Die Zahl der Nachfolger dieser Knoten ist die Stelligkeit des Symbols. Konstanten, Variablen und Aussagensymbole bilden Blätter. Runde Knoten: logische Symbole Eckige Knoten: verfügbare Symbole	

Zum Selbststudium

Darüber hinaus:

- Häufig ist es nützlich, zweistellige Funktionssymbole oder Prädikatensymbole zwischen die zugehörigen Terme zu schreiben. In der Mathematik ist dies z.B. die Standardkonvention für Operationssymbole für Addition, Multiplikation etc. und Relationen wie Gleichheit, größer, kleiner etc. Um diese Notation, die sog. Infixschreibweise, zuzulassen, muss die Syntaxdefinition der Logik-Sprache geeignet geändert werden.

Prädikate und Argument-Reihenfolge

Hamburg	ist eine Stadt					
hh	S1					
	hh					S1(hh)

Hamburg	liegt nördlich von	München		
hh	N	m		
	hh, m			N(hh, m)

Hamburg	ist eine Stadt		nördlich von	München	
(Hamburg	ist eine Stadt	und	liegt nördlich von	München)	
hh	S1	∧	N	m	
	hh		hh, m		S1(hh) ∧ N(hh, m)

Hamburg	ist eine Stadt,	die	nördlich von	München	liegt	
hh	S1		N	m		
	hh		hh, m			S1(hh) ∧ N(hh, m)

ersetzt 9-18

$\exists x\, S1(x)$

Es gibt	eine Stadt
(Es gibt etwas,	das ist eine Stadt)
∃	S1
x	x

$\exists x\, (S1(x) \wedge G(x))$

Es gibt	eine große		Stadt
(Es gibt etwas,	das ist groß	und	eine Stadt)
∃	G	∧	S1
x	x		x

$\exists x\, (S1(x) \wedge N(x, m))$

Eine	Stadt		liegt nördlich von	München
(Es gibt etwas,	das ist eine Stadt	und	liegt nördlich von	München.)
∃	S1	∧	N	m
x	x		x, m	

ersetzt 9-18

Übersetzungen Deutsch → PL: Existenzquantor (2)

$\neg\exists x\ E1(x)$

Es gibt kein	Einhorn				
(Es gibt nichts, das ein Einhorn ist)					
$\neg\exists$	E1				
x	x				

$\neg\exists x\ (S1(x) \wedge N(x, m))$

Keine	Stadt		liegt nördlich von	München	
(Es gibt nichts,	das eine Stadt ist	und	nördlich von	München	liegt.)
$\neg\exists$	S1	$\wedge$	N	m	
x	x		x, m		

$\neg\exists x\ (S1(x) \wedge N(x, x))$

Keine	Stadt		liegt nördlich von	sich selbst	
(Es gibt nichts,	das eine Stadt ist	und	nördlich von	sich selbst	liegt.)
$\neg\exists$	S1	$\wedge$	N		
x	x		x, x		

ersetzt 9-18

Übersetzungen Deutsch → PL: Allquantor

$\forall x\ T(x)$

Alles	ist toll
(Für jedes gilt:	es ist toll)
$\forall$	T
x	x

$\forall x\ (S1(x) \Rightarrow G(x))$

Jede	Stadt		ist groß
(wenn etwas	eine Stadt ist,	dann	ist es groß)
(Für jedes gilt:	wenn es eine Stadt ist,	dann	ist es groß)
$\forall$	S1	$\Rightarrow$	G
x	x		x

$\forall x\ (S1(x) \Rightarrow N(x, m))$

Jede	Stadt		liegt nördlich von	München
(wenn etwas	eine Stadt ist,	dann	liegt es nördlich von	München)
(Für jedes gilt:	wenn es eine Stadt ist,	dann	liegt es nördlich von	München)
$\forall$	S1	$\Rightarrow$	N	m
x	x		x, m	

Übersetzungen Deutsch → PL (3)

Beispiel: $\forall x\ (M1(x) \Rightarrow \exists y\ M2(y,x))$

Jeder	Mensch		hat eine	Mutter
(Wenn etwas	ein Mensch ist,	dann	hat er eine	Mutter)
(Für jedes gilt:	wenn es ein Mensch ist,	dann	hat es eine	Mutter)
$\forall$	M1	$\Rightarrow$	$\exists$	M2
x	x		y	y, x

Aber:

Jeder / jede hat eine Mutter bzw. Alle haben eine Mutter

werden auch übersetzt in: $\forall x\ (M1(x) \Rightarrow \exists y\ M2(y,x))$

denn:

- das Deutsche stellt *sortale* Bedingungen daran, welche Arten von Entitäten in der Beziehung ‚_ ist Mutter von _' stehen können, damit der entsprechende Satz *sinnvoll* ist.
- die Standard-Prädikatenlogik stellt keine entsprechenden Bedingungen an die Variablen

➔ Sortale Bedingungen müssen in der Standard-Prädikatenlogik explizit durch einstellige Prädikatensymbole formuliert werden.

z.B.: Jeder / jede → M1(x)

➔ Sortenlogik ≈ Prädikatenlogik mit sortierten Variablen
[wird in FGI-3 (Masterstudium) behandelt]

Zum Selbststudium: Sortenlogik

Sorten

- ermöglichen es, Kategorien der Domäne zu berücksichtigen und hierbei auch unterschiedliche Rollen, die Entitäten der Domäne spielen, einzubeziehen, z.B.
  in Flugverb(ab8862, ham, zrh, 240507, 06:30, 07:50)
  - treten vier Sorten auf: 'flightnumber', 'airport', 'date', 'time'
  - die Argumente der Sorte
    - 'airport' tritt in zwei Rollen auf: 'origin' und 'destination'
    - 'time' tritt in zwei Rollen auf: 'departure' und 'arrival'
- Entsprechend differenzierte Analysen sind Grundlage für
  - die Strukturierung einer Domäne in Relationalen Datenbanken → Modul GDB
  - die Darstellung von Regularitäten in Wissensbasierten Systemen (& Deduktiven Datenbanken) → Modul GWV
- ermöglichen es, gewisse Schritte des automatischen Beweisens effizienter zu realisieren.
  [wird – nicht regelmässig – in FGI-3 (Masterstudium) behandelt]

Funktions- und Prädikatensymbole

Welche Übersetzung

‚ist (die) Hauptstadt von‘	→	Hs	zweistelliges Prädikatensymbol	?
‚Hauptstadt von‘	→	hs	einstelliges Funktionssymbol	

➔ Bedingung:
Mit einem Funktionssymbol gebildete Terme benennen eindeutig ein Objekt.

Beispiele

	Darstellung mit einstelligem Funktionssymbol	Darstellung mit zweistelligem Prädikatensymbol
Die Hauptstadt von Frankreich liegt an der Seine.	Liegt_an(hs(fr), s)	∃x (Liegt_an(x, s) ∧ Hs(x, fr))
Paris ist die Hauptstadt von Frankreich.	p = hs(fr)	Hs(p, fr)

➔ Statt einstelliger Funktionssymbole können auch zweistellige Prädikatensymbole verwendet werden, aber: dies führt zu anderen prädikatenlogischen Darstellungen.

➔ Bei Verwendung von Funktionssymbolen ist ein zusätzliches logisches Symbol = (Identität) nützlich. ➔ Prädikatenlogik mit Identität (nicht Gegenstand dieser Vorlesung, wird in FGI-3 behandelt.)

Funktions- und Prädikatensymbole (2)

	Funktionale Schreibweise	Relationale Schreibweise
Paris ist die Hauptstadt von Frankreich.	p = hs(fr)	Hs(p, fr)
	Fordert für jede Konstante a die Eindeutigkeit des Funktionswertes hs(a).	Ermöglicht die Existenz mehrerer Objekte b_1 und b_2, die zu einer Konstante a in der "Hauptstadt-Relation" Hs stehen können: $Hs(b_1, a)$ und x.

Jede n-stellige Funktion kann auch als (n+1)-stellige Relation dargestellt werden.

- Dabei geht die explizite Funktionale Abhängigkeit verloren.
- In Relationalen Datenbanken spielt "Funktionale Abhängigkeit" eine wichtige Rolle.
  → Kap. 7. Logischer DB-Entwurf im Modul GDB

Freie Variablen einer Formel

Definition 9.4 (freie Variablen von Termen und Formeln)

Wir definieren eine Abbildung FV von Termen und Formeln auf Mengen von Variablen rekursiv wie folgt:

$FV(a) = \{\ \}$	für jede Konstante a
$FV(x) = \{ x \}$	für jede Variable x
$FV(f(t_1, \ldots, t_k)) = FV(t_1) \cup \ldots \cup FV(t_k)$	für jedes k-stellige Funktionssymbol f und alle Terme $t_1, \ldots, t_k$
$FV(A) = \{\ \}$	für jedes Aussagensymbol A
$FV(P(t_1, \ldots, t_k)) = FV(t_1) \cup \ldots \cup FV(t_k)$	für jedes k-stellige Prädikatensymbol P und alle Terme $t_1, \ldots, t_k$
$FV(\neg F) = FV(F)$	für jede Formel F
$FV((F \wedge G)) = FV((F \vee G)) = FV((F \Rightarrow G)) = FV((F \Leftrightarrow G)) = FV(F) \cup FV(G)$	für beliebige Formeln F und G
$FV(\exists x\, F) = FV(\forall x\, F) = FV(F) \setminus \{ x \}$	für jede Formel F und jede Variable x

Für jede Formel F nennen wir die Elemente von $FV(F)$ die *in F frei vorkommenden Variablen*.

Gebundenes und freies Vorkommen von Variablen

Definition 9.5 (Skopus / Gebundene / freie Variable)

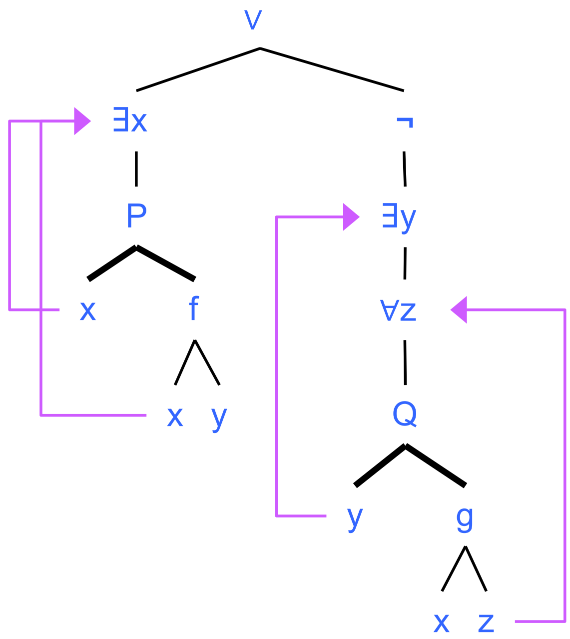
- Zur Erinnerung: Wenn F eine Formel und x eine Variable ist, dann sind auch $\exists x\, F$ und $\forall x\, F$ Formeln (s. Def. 9.3.4).
  Bezüglich der Formeln $\exists x\, F$ und $\forall x\, F$ bezeichnen wir F auch als den Skopus des Quantors und x als die Quantorenvariable.
- Eine Variable x, die *im Skopus* eines Quantors ($\exists x$ bzw. $\forall x$) mit der Quantorenvariable x frei vorkommt, ist durch den Quantor in dieser Position *gebunden*.
- Eine Variable x, die als Teilterm in einer Formel F enthalten aber in dieser Position durch keinen Quantor gebunden ist, ist in dieser Position frei in F.
- Eine Formel ohne freies Vorkommen von Variablen heißt *geschlossen*.
  Geschlossene Formeln werden auch als prädikatenlogische Aussagen bezeichnet.

Beispiel: Variablenbindung durch Quantoren

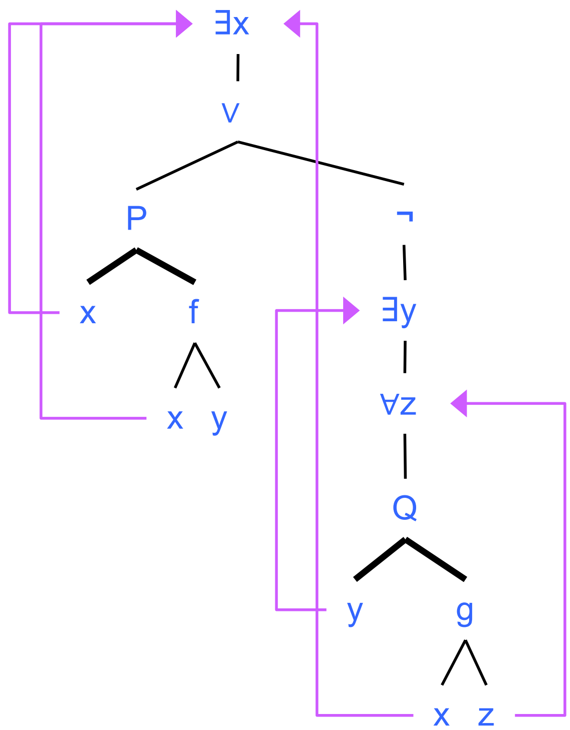
$$\underbrace{\forall x}_{\text{Quantor}}\ \underbrace{(P(x) \vee R(x, y))}_{\text{Skopus}}$$

gebunden frei

Beispiel: ∃x P(x, f(x, y)) ∨ ¬∃y ∀z Q(y, g(x, z))

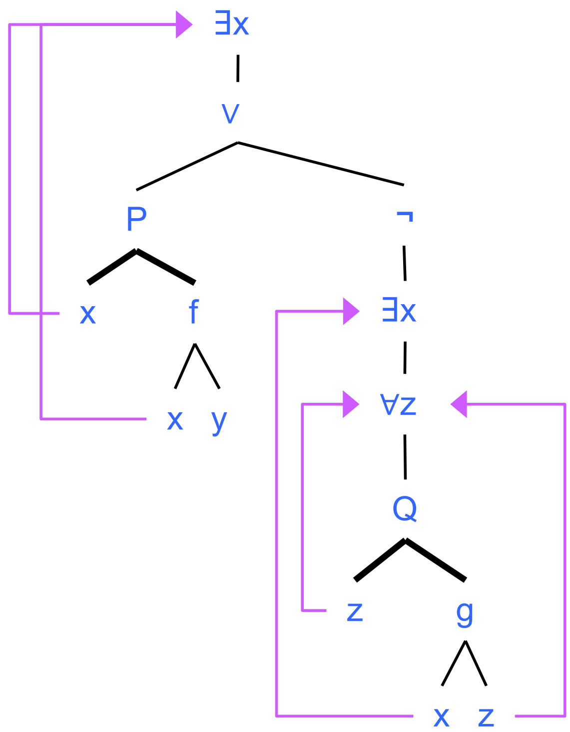


Beispiel: ∃x (P(x, f(x, y)) ∨ ¬∃y ∀z Q(y, g(x, z)))



ergänzt hinter 9-26

Beispiel: ∃x (P(x, f(x, y)) ∨ ¬∃x ∀z Q(z, g(x, z)))



Semantik: Aussagenlogik – Prädikatenlogik

Auswertung von Formeln in der Aussagenlogik ($\mathcal{L}_{AL}$)

- Formeln werden auf Wahrheitswerte abgebildet.
- Aussagensymbole sind die frei interpretierbaren Bestandteile (verfügbare Symbole).
- Aussagensymbolen werden durch Belegungen ausgewertet.
- Die Auswertung komplexer Formeln bestimmt sich aus einer Belegung und festen Regeln für die Auswertung der Junktoren.

Auswertung von Ausdrücken in der Prädikatenlogik ($\mathcal{L}_{PL}$)

- Terme werden auf (beliebige) Objekte einer Grundmenge abgebildet.
- Zusätzliche frei interpretierbare Bestandteile (verfügbare Symbole):
  (freie) Variablen, Konstanten, Funktionssymbole und Prädikatensymbole
- Sie werden durch Interpretationen bezüglich einer Grundmenge (Domäne, Universum) ausgewertet.
- Zusätzliche Regeln für die Auswertung von komplexen Termen, atomaren Formeln und für komplexe Formeln mit Quantoren
- Quantoren beeinflussen die Auswertung der von ihnen gebundenen Variablen.

Semantik: Aussagenlogik – Prädikatenlogik (Forts.)

- Die verfügbaren Symbole der prädikatenlogischen Ausdrücke werden mit Hilfe der Grundmenge (Domäne, Universum) interpretiert:
  - freie Variablen und Konstanten durch Objekte der Grundmenge
  - Funktionssymbole durch Funktionen über der Grundmenge
  - Prädikatensymbole durch Mengen über der Grundmenge
  - Aussagensymbole durch Wahrheitswerte.
  - Komplexe Ausdrücke (Terme, Formeln) werden hierauf aufbauend ausgewertet.

➔ Auch in der Semantik der Prädikatenlogik spielen Wahrheitswerte eine zentrale Rolle.

➔ Wahrheitswerte atomarer Formeln ergeben sich durch die Interpretation über der Grundmenge (Domäne).

Semantik der Prädikatenlogik: Strukturen

Definition 9.6 (Struktur)

Eine *Struktur* ist ein Paar $\mathcal{A} = (\mathsf{U}, \mathsf{I})$, wobei

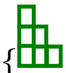
- U *(Universum, Grundmenge, Domäne)* eine beliebige, nicht leere Menge ist und
- I (*Interpretation*, *Auswertung*, *valuation*) eine Abbildung ist.
  - Der Definitionsbereich von I ist $\mathcal{V}_{\mathrm{PL}}$ (Vokabular der Prädikatenlogik).

    I bildet
    - Variablen und Konstanten auf Elemente der Grundmenge U,
    - k-stellige Funktionssymbolen auf k-stellige Funktionen über U,
    - k-stellige Prädikatensymbole auf Mengen von k-Tupeln über U und
    - Aussagensymbole auf Wahrheitswerte ab.

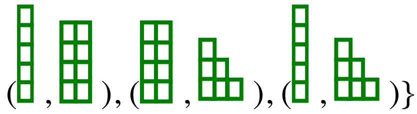
- Wenn mehrere Strukturen $\mathcal{A}$ und $\mathcal{B}$ bzw. $\mathcal{A}_1$ und $\mathcal{A}_2$, gleichzeitig betrachtet werden sollen, schreiben wir auch $\mathcal{A} = (\mathsf{U}_{\mathcal{A}}, \mathsf{I}_{\mathcal{A}})$ und $\mathcal{B} = (\mathsf{U}_{\mathcal{B}}, \mathsf{I}_{\mathcal{B}})$; bzw. $\mathcal{A}_1 = (\mathsf{U}_1, \mathsf{I}_1)$ und $\mathcal{A}_2 = (\mathsf{U}_2, \mathsf{I}_2)$, um die Universen und Interpretationen auseinanderzuhalten.

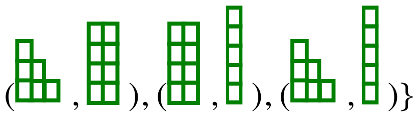
Beispiel (1): Interpretationen für $\mathcal{L}_{PL}$

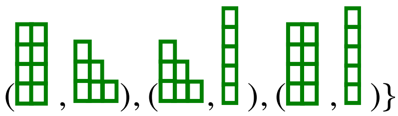
Konstanten: a, b, c   Variable: x
Prädikatensymbol: größer (2-stellig)

Die Domäne U = {  ,  ,  }

Relationen über der Domäne U:

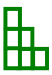
H = 

B = 

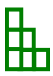
V = 

Beispiel (1): Interpretationen für $\mathcal{L}_{PL}$ (Forts. – 1)

$\mathcal{A}_1 = (U, I_1)$

I_1(a) =  I_1(b) =  I_1(c) =  I_1(x) =  I_1(größer) = H

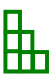
$\mathcal{A}_2 = (U, I_2)$

I_2(a) =  I_2(b) =  I_2(c) =  I_2(x) =  I_2(größer) = H

:

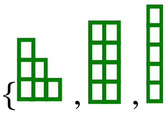
:

$\mathcal{A}_4 = (U, I_4)$

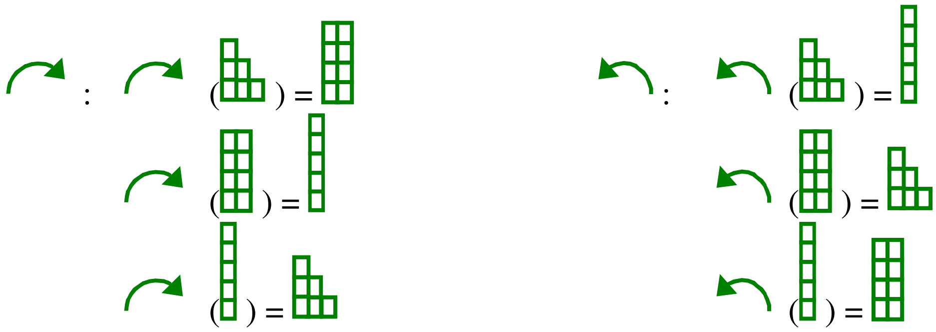
I_4(a) =  I_4(b) =  I_4(c) =  I_4(x) =  I_4(größer) = V

Beispiel (1): Interpretationen für $\mathcal{L}_{PL}$ (Forts. – 2)

Konstanten: a, b, c   Variable: x
Prädikatensymbol: größer (2-stellig)
Funktionssymbol: nachfolger (1-stellig)

Die Domäne U = {  }

Funktionen über der Domäne U:



Tabellarische Darstellung verschiedener Interpretationen über einer Domäne

		a	b	c	x	größer	nachfolger	…
$\mathcal{A}_1$	I_1					H		
$\mathcal{A}_2$	I_2					H		
	:							
$\mathcal{A}_4$	I_4					V		
$\mathcal{A}_5$	I_5					H		

Entsprechend zu Wahrheitstafeln.
Aber: Es gibt unendlich viele Interpretationen, die Tafel kann also nicht vollständig werden.
Bei den Ausdrücken unseres Vokabulars stimmen $\mathcal{A}_1$ und I_1 immer genau überein.

Semantik der Prädikatenlogik: Komplexe Terme und Formeln

Definition 9.7 (Auswertung komplexer Terme und atomarer Formeln)

Sei F eine Formel und $\mathcal{A} = (U, I)$ eine Struktur.

I wird rekursiv zu einer Auswertung fortgesetzt, für die wir dann einfach $\mathcal{A}$ schreiben.

- Rekursive Definition von $\mathcal{A}$ für Terme:

T*-1 Für jede Variable x ist: $\mathcal{A}(x) = I(x)$.

T*-2 Für jede Konstante a ist: $\mathcal{A}(a) = I(a)$.

T*-3 Für jede Liste von k Termen $t_1, \ldots, t_k$ und jedes k-stellige Funktionssymbol ist:

$$\mathcal{A}(f(t_1, \ldots, t_k)) = I(f)(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k))$$

  - $I(f)$ ist eine k-stellige Funktion über U. Diese Funktion wird auf die Liste von Objekten angewendet, die bei der Auswertung der Terme entsteht.

- Rekursive Definition von $\mathcal{A}$ für atomare Formeln:

F*-1 Für jedes Aussagensymbol A ist: $\mathcal{A}(A) = I(A)$

F*-2 Für jede Liste von k Termen $t_1, \ldots, t_k$ und jedes k-stellige Prädikatensymbol P ist:

$$\mathcal{A}(P(t_1, \ldots, t_k)) = \begin{cases} \mathbf{1}, \text{ falls } (\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k)) \in I(P) \\ \mathbf{0}, \text{ sonst} \end{cases}$$

Zum Selbststudium: Alternative Schreibweise

U. Schöning (vgl. Abschnitt 2.1) verwendet eine „Abkürzende Schreibweise“:

$P^{\mathcal{A}}$ statt $I(P)$, $f^{\mathcal{A}}$ statt $I(f)$, $x^{\mathcal{A}}$ statt $I(x)$

- Damit ergibt sich dann z.B.:

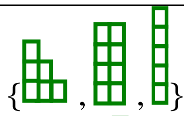
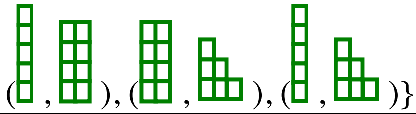
T*-3 Für jede Liste von k Termen $t_1, \ldots, t_k$ und jedes k-stellige Funktionssymbol ist:

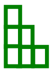
$$\mathcal{A}(f(t_1, \ldots, t_k)) = f^{\mathcal{A}}(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k))$$

F*-2 Für jede Liste von k Termen $t_1, \ldots, t_k$ und jedes k-stellige Prädikatensymbol P ist:

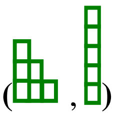
$$\mathcal{A}(P(t_1, \ldots, t_k)) = \begin{cases} \mathbf{1}, \text{ falls } (\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k)) \in P^{\mathcal{A}} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

Beispiel (2): Auswertung für $\mathcal{L}_{PL}$

Das Vokabular	$\mathcal{A}_1$ = (U, I_1)
Konstanten: a, b, c Prädikatensymbol: größer (2-stellig) Funktionssymbol: nachfolger (1-stellig)	Die Domäne U =  H = 

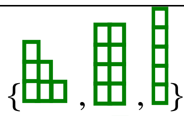
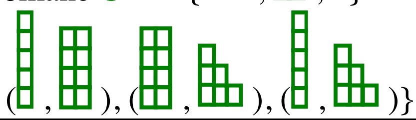
	a	b	c	größer	nachfolger	größer(a, c)	größer(b, a)
$\mathcal{A}_1$				H		??	

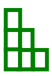
$\mathcal{A}_1$(größer(a, c)) = ?

($\mathcal{A}_1$(a), $\mathcal{A}_1$(c)) =  $\notin$ H

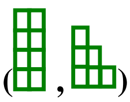
also: $\mathcal{A}_1$(größer(a, c)) = **0**

Beispiel (2): Auswertung für $\mathcal{L}_{PL}$ (Forts. – 1)

Das Vokabular	$\mathcal{A}_1$ = (U, I_1)
Konstanten: a, b, c Prädikatensymbol: größer (2-stellig) Funktionssymbol: nachfolger (1-stellig)	Die Domäne U =  H = 

	a	b	c	größer	nachfolger	größer(a, c)	größer(b, a)
$\mathcal{A}_1$				H		**0**	??

$\mathcal{A}_1$(größer(b, a)) = ?

($\mathcal{A}_1$(b), $\mathcal{A}_1$(a)) =  $\in$ H

also: $\mathcal{A}_1$(größer(b, a)) = **1**

überarbeitet

Beispiel (2): Auswertung für $\mathcal{L}_{PL}$ (Forts. – 2a)

Das Vokabular	$\mathcal{A}_1$ = (U, I_1)
Konstanten: a, b, c Prädikatensymbol: größer (2-stellig) Funktionssymbol: nachfolger (1-stellig)	Die Domäne U = { , , }

	a	b	c	größer	nachfolger	nachfolger(a)	nachfolger(c)
$\mathcal{A}_1$				H		??	??

$\mathcal{A}_1$(nachfolger(a)) = ?

= $\mathcal{A}_1$(nachfolger)($\mathcal{A}_1$(a))

= ( ) =

$\mathcal{A}_1$(nachfolger(c)) = ?

= $\mathcal{A}_1$(nachfolger)($\mathcal{A}_1$(c))

= ( ) =

ergänzt hinter 9-38

Beispiel (2): Auswertung für $\mathcal{L}_{PL}$ (Forts. – 2b)

Das Vokabular	$\mathcal{A}_1$ = (U, I_1)
Konstanten: a, b, c Prädikatensymbol: größer (2-stellig) Funktionssymbol: nachfolger (1-stellig)	Die Domäne U = { , , } H = { (,), (,), (,)}

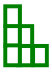
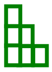
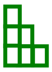
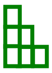
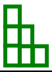
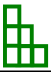
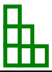
	a	b	c	größer	nachfolger	nachfolger(a)	nachfolger(c)	größer(nachfolger(a), nachfolger(c))
$\mathcal{A}_1$				H				??

$\mathcal{A}_1$(größer(nachfolger(a), nachfolger(c))) = ?

($\mathcal{A}_1$(nachfolger(a)), $\mathcal{A}_1$(nachfolger(c))) = (,) $\in$ H

also: $\mathcal{A}_1$(größer(nachfolger(a), nachfolger(c))) = 1

Zum Selbstausfüllen

	a	b	c	größer	nachfolger	größer (a, c)	größer (b, a)	nachfolger (a)	nachfolger (c)	größer(nachfolger(a), nachfolger(c))
$\mathcal{A}_1$				H		0	1			1
$\mathcal{A}_2$				H						
$\mathcal{A}_4$				V						
$\mathcal{A}_5$				H						

Semantik der Prädikatenlogik: Auswertung komplexer Formeln

Definition 9.8 (Auswertung komplexer Formeln ohne Quantoren)

Fortsetzung der rekursiven Definition der Auswertung von Formeln (genauso wie in der Aussagenlogik):

F*-3 Für alle Formeln F, G, und alle Strukturen $\mathcal{A}$, ist

$$\mathcal{A}(\neg F) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}(F) = \mathbf{0} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}((F \wedge G)) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}(F) = \mathbf{1} \text{ und } \mathcal{A}(G) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}((F \vee G)) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}(F) = \mathbf{1} \text{ oder } \mathcal{A}(G) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}((F \Rightarrow G)) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}(F) = \mathbf{0} \text{ oder } \mathcal{A}(G) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}((F \Leftrightarrow G)) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}(F) = \mathcal{A}(G) \\ \mathbf{0}, \text{ sonst} \end{cases}$$

Semantik der Prädikatenlogik: x-Varianten

Definition 9.9 (x-Varianten von Strukturen)

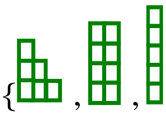
Sei $\mathcal{A} = (\mathsf{U}, \mathsf{I})$ eine Struktur, $\mathsf{d} \in \mathsf{U}$, und x eine Variable.
$\mathcal{A}_{[\mathsf{x}/\mathsf{d}]} = (\mathsf{U}, \mathsf{I}_{[\mathsf{x}/\mathsf{d}]})$ ist diejenige Struktur, die x als d interpretiert und ansonsten komplett mit $\mathcal{A}$ übereinstimmt. Es gilt also für alle verfügbaren Symbole τ:

$$\mathcal{A}_{[\mathsf{x}/\mathsf{d}]}(\tau) = \mathsf{I}_{[\mathsf{x}/\mathsf{d}]}(\tau) = \begin{cases} \mathsf{d}, \text{ falls } \tau = \mathsf{x} \\ \mathsf{I}(\tau), \text{ sonst} \end{cases}$$

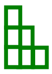
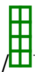
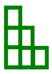
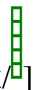
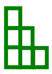
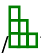
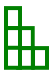
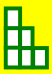
- Weicht eine Struktur, wie die hier gebildete, höchstens bezüglich der Interpretation einer Variable x von der Struktur $\mathcal{A}$ ab, dann heißt sie x-Variante zu $\mathcal{A}$.

Beispiel (3): x-Varianten

Variable: x

Domäne U = {  }

Bei endlichen Domänen lassen sich alle Varianten in einer Tabelle auflisten.
Variiert werden nur Variablen.

	a	b	c	x	größer	nachfolger
$\mathcal{A}_1$					H	
$\mathcal{A}_{1[\mathsf{x}/}$  $]$					H	
$\mathcal{A}_{1[\mathsf{x}/}$  $]$					H	
$\mathcal{A}_{1[\mathsf{x}/}$  $]$					H	

$\mathcal{A}_1 = \mathcal{A}_{1[\mathsf{x}/}$  $]$ Es ist aber unschädlich, sich in solchen Tabellen zu wiederholen.

Semantik der Prädikatenlogik: Formeln mit Quantoren

Definition 9.10 (Auswertung komplexer Formeln mit Quantoren):

F*-4 Für jede Variable x, jede Formel F und jede Struktur $\mathcal{A}$ ist

$$\mathcal{A}(\forall x\ F) = \begin{cases} \mathbf{1}, \text{ falls für alle } d \in U \text{ gilt: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}(\exists x\ F) = \begin{cases} \mathbf{1}, \text{ falls es ein } d \in U \text{ gibt mit: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

➔ Auf den Skopus des Quantors werden alle Varianten der Auswertung angewendet.
➔ Bei der Auswertung einer quantifizierten Formel ist der ursprüngliche Wert der Quantorenvariable unerheblich.
➔ Die Werteberechnung kann nicht nur ‚von unten nach oben' erfolgen.

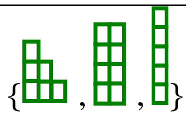
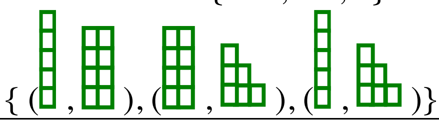
Dieses schließt die Auswertung der prädikatenlogischen Formeln ab.

Beobachtung zu Definitionen 9.7, 9.8 und 9.10

Stimmen zwei Strukturen bzgl. des Universums und bzgl. der Interpretationen aller verfügbaren Symbole und freien Variablen der Formel überein, dann liefern sie auch für die Formel denselben Wert.

überarbeitet

Beispiel (4): Auswertung von Formeln mit Quantoren

Das Vokabular	
Konstanten: b Prädikatensymbol: größer (2-stellig) Variable: x	$\mathcal{A}_1 = (U, I_1)$ Die Domäne U =  H = 

	b	x	größer	größer(x, b)	∃x größer(x, b)
$\mathcal{A}_1$			H		??

$\mathcal{A}_1(\exists x\ \text{größer}(x, b))$ = ?

$\mathcal{A}_{1[x/\ ]}$			H	**0**
$\mathcal{A}_{1[x/\ ]}$			H	**1**
$\mathcal{A}_{1[x/\ ]}$			H	**0**

$$\mathcal{A}(\exists x\ F) = \begin{cases} \mathbf{1}, \text{ falls es ein } d \in U \text{ gibt mit: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

➔ $\mathcal{A}_1(\exists x\ \text{größer}(x, b)) = \mathbf{1}$ denn es gibt ein $d \in U$ mit: $\mathcal{A}_{1[x/d]}(\text{größer}(x, b)) = \mathbf{1}$

Beispiel (4): Auswertung von Formeln mit Quantoren (Forts. – 1)

Das Vokabular	$\mathcal{A}_1 = (U, I_1)$
Konstanten: b Prädikatensymbol: größer (2-stellig) Variable: x	Die Domäne U = { , , } H = { (,), (,), (,)}

	b	x	größer	größer(x, b)	∀x größer(x, b)
$\mathcal{A}_1$			H		??

$\mathcal{A}_1$(∀x größer(x, b)) = ?

$\mathcal{A}_{1[x/\ ]}$			H	0
$\mathcal{A}_{1[x/\ ]}$			H	1
$\mathcal{A}_{1[x/\ ]}$			H	0

$$\mathcal{A}(\forall x\ F) = \begin{cases} \mathbf{1}, \text{ falls für alle } d \in U \text{ gilt: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

➔ $\mathcal{A}_1$(∀x größer(x, b)) = **0** denn es gilt nicht für alle d ∈ U: $\mathcal{A}_{1[x/d]}$ (größer(x, b)) = **1**

Beispiel (4): Tabellenform

Variable: x

Domäne U = { , , }

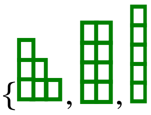
$$\mathcal{A}(\forall x\ F) = \begin{cases} \mathbf{1}, \text{ falls für alle } d \in U \text{ gilt: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

$$\mathcal{A}(\exists x\ F) = \begin{cases} \mathbf{1}, \text{ falls es ein } d \in U \text{ gibt mit: } \mathcal{A}_{[x/d]}(F) = \mathbf{1} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

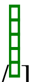
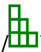
	b	c	x	größer	größer (x, b)	∃x größer (x, b)	∀x größer (x, b)	größer (x, c)	∃x größer (x, c)	¬größer(x, c)	∀x ¬größer(x, c)
$\mathcal{A}_1$				H		1	0		0		1
$\mathcal{A}_{1[x/\ ]}$				H	0			0		1	
$\mathcal{A}_{1[x/\ ]}$				H	1			0		1	
$\mathcal{A}_{1[x/\ ]}$				H	0			0		1	

Beispiel (4): Tabellenform (Forts. – 1)

Variable: x

Domäne U = {  }

Beachte: Für alle Strukturen $\mathcal{A}$, Variablen x, Formeln F und d, e ∈ U gilt: $\mathcal{A}_{[x/d][x/e]} = \mathcal{A}_{[x/e]}$ (s. Def. 9.9, 9.10)
$\mathcal{A}_{[x/d]}(\forall x\ F) = \mathcal{A}(\forall x\ F)$ und $\mathcal{A}_{[x/d]}(\exists x\ F) = \mathcal{A}(\exists x\ F)$

	b	c	x	größer	größer(x, b)	∃x größer(x, b)	∀x größer(x, b)	größer(x, c)	∃x größer(x, c)	¬größer(x, c)	∀x ¬größer(x, c)
$\mathcal{A}_1$				H	0	1	0	0	0	1	1
$\mathcal{A}_{1[x/}$$]$				H	0	1	0	0	0	1	1
$\mathcal{A}_{1[x/}$$]$				H	1	1	0	0	0	1	1
$\mathcal{A}_{1[x/}$$]$				H	0	1	0	0	0	1	1

ergänzt hinter 9-47

Interpretation: Variablen und Quantoren

Freie Variablen

- verhalten sich bei der Interpretation wie Konstanten.
- $\mathcal{A}_1$(größer(x, b)): In der Bestimmung des Wahrheitswertes wird x durch $\mathcal{A}_1$ ausgewertet.

Gebundene Variablen

- Der bindende Quantor schirmt die Variable gewissermaßen gegen die Außenwelt ab. Die zu verwendende Interpretation wird durch den Quantor variiert.
- $\mathcal{A}_1$(∃x größer(x, b)): die Interpretation der Variable x wird systematisch variiert und der Skopus größer(x, b) durch alle Varianten von $\mathcal{A}_1$ ausgewertet.
- Über den Quantor werden die Ergebnisse der Variation des Variablenwertes für den Skopus (größer(x, b)) zusammengefaßt.
- Quantoren schaffen einen neuen Namensraum für die Quantorenvariable.

Gemischte Vorkommen

- Tritt in einer Formel dieselbe Variable frei und gebunden auf, dann werden sie in verschiedenen Positionen unterschiedlich interpretiert.
- Nutzen zwei Quantoren dieselbe Variable, dann erfolgt die Variation unabhängig voneinander.

Modell, Gültigkeit, Unerfüllbarkeit

Definition 9.11 (ganz analog zur Aussagenlogik)

- Eine Struktur $\mathcal{A}$ heißt genau dann *Modell* für F, wenn gilt:
  $\mathcal{A}(\mathsf{F}) = \mathbf{1}$
- In diesen Fällen sagen und schreiben wir auch:
  - $\mathcal{A}$ macht F wahr
  - F gilt in $\mathcal{A}$
  - $\mathcal{A} \models \mathsf{F}$
- Eine Struktur $\mathcal{A}$ ist ein *Modell* für eine Formelmenge **M** GDW. $\mathcal{A}$ ein *Modell* für jede Formel $\mathsf{F} \in \mathbf{M}$ ist.

- Eine Formel F (eine Formelmenge **M**) heißt *erfüllbar*, falls ein Modell $\mathcal{A}$ für F (bzw. **M**) existiert, anderenfalls heißt F (bzw. **M**) *unerfüllbar*.
- Eine Formel F (eine Formelmenge **M**) heißt *falsifizierbar*, falls eine Struktur $\mathcal{A}$ kein Modell für F (bzw. **M**) ist.
- Eine Formel F (eine Formelmenge **M**) heißt *gültig*, falls jede Struktur $\mathcal{A}$ ein Modell für F (bzw. **M**) ist.
  - F ist *logisch wahr*
  - $\models \mathsf{F}$
  - Die Definition der Gültigkeit beruht auf *allen Strukturen mit beliebigen Domänen und Interpretationen*.

Äquivalenz, Folgerung

Definition 9.12 (ganz analog zur Aussagenlogik)

- Zwei Formeln G und F sind *(logisch) äquivalent*, falls für jede Struktur $\mathcal{A}$ gilt:
  $\mathcal{A}(\mathsf{F}) = \mathcal{A}(\mathsf{G})$.
- Eine Formel G *folgt (logisch)* aus einer Formel F (einer Formelmenge **M**) falls für jede Struktur $\mathcal{A}$ gilt:
  Wenn $\mathcal{A}$ ein Modell für F (bzw. **M**) ist, dann ist $\mathcal{A}$ auch ein Modell für G.
  - $\mathsf{F} \models \mathsf{G}$ bzw. $\mathbf{M} \models \mathsf{G}$

Gültigkeit in der Prädikatenlogik (1)

Beispiele

- $\forall x\, P(x) \vee \neg\forall x\, P(x)$
  - ist das Resultat der Substitution mit $sub(A) = \forall x\, P(x)$, angewendet auf die aussagenlogische Tautologie $(A \vee \neg A)$.

Satz 9.13

Wenn in einer aussagenlogischen Tautologie prädikatenlogische Formeln (uniform) substituiert werden, ergibt dies eine gültige Formel der Prädikatenlogik.
ohne Beweis, aber vgl. Satz 6.2

- $\forall x\, (P(x) \vee \neg P(x))$
  - ist nicht das Resultat eine Substitution einer aussagenlogischen Tautologie.
  - ist aber eine gültige Formel der Prädikatenlogik. (ist noch zu zeigen!!!!)
- $\forall x\, P(x) \Rightarrow \exists y\, P(y)$
  - ist eine gültige Formel der Prädikatenlogik. (ist noch zu zeigen!!!!)

Gültigkeit in der Prädikatenlogik (2)

Gültigkeit von $\forall x\, (P(x) \vee \neg P(x))$

Beweis

Vorbemerkung: Nach Def.9.8 gilt für alle Strukturen $\mathcal{A}'$:

$$\mathcal{A}'(P(x) \vee \neg P(x)) = \begin{cases} \mathbf{1}, \text{ falls } \mathcal{A}'(P(x)) = \mathbf{1} \text{ oder } \mathcal{A}'(P(x)) = \mathbf{0} \\ \mathbf{0}, \text{ sonst} \end{cases}$$

Also: $\mathcal{A}'(P(x) \vee \neg P(x)) = \mathbf{1}$, (maW: $P(x) \vee \neg P(x)$ ist eine Tautologie).

Es sei $\mathcal{A} = (U, I)$ eine beliebige Struktur.
Die x-Varianten von $\mathcal{A}$ liefern (entsprechend der Vorbemerkung) für diese Formel alle den Wahrheitswert $\mathbf{1}$.
Entsprechend Def. 9.10 ist dann $\mathcal{A}(\forall x\, (P(x) \vee \neg P(x))) = \mathbf{1}$.
Da die Wahl von $\mathcal{A}$ nicht eingeschränkt war, ist also $\forall x\, (P(x) \vee \neg P(x))$ gültig.

Gültigkeit in der Prädikatenlogik (3)

Gültigkeit von $\forall x\ P(x) \Rightarrow \exists y\ P(y)$

Beweis
Es sei $\mathcal{A} = (U, I)$ eine beliebige Struktur.
Wenn $\mathcal{A}(\forall x\ P(x) \Rightarrow \exists y\ P(y)) = \mathbf{0}$ ist, dann ist $\mathcal{A}(\forall x\ P(x)) = \mathbf{1}$ und $\mathcal{A}(\exists y\ P(y)) = \mathbf{0}$.
$\mathcal{A}(\exists y\ P(y)) = \mathbf{0}$ kann nur gelten, wenn für jedes $d \in U$ die y-Variante $\mathcal{A}_{[y/d]}$ von $\mathcal{A}$ folgendes liefert: $\mathcal{A}_{[y/d]}(P(y)) = \mathbf{0}$, also $\mathcal{A}_{[y/d]}(y) = d \notin \mathcal{A}_{[y/d]}(P) = \mathcal{A}(P)$. Dafür muss $\mathcal{A}(P) = \emptyset$ sein.
$\mathcal{A}(\forall x\ P(x)) = \mathbf{1}$ kann nur gelten, wenn für jedes $e \in U$ die x-Variante $\mathcal{A}_{[x/e]}$ von $\mathcal{A}$ folgendes liefert: $\mathcal{A}_{[x/e]}(P(x)) = \mathbf{1}$, also $\mathcal{A}_{[x/e]}(x) = e \in \mathcal{A}_{[x/e]}(P) = \mathcal{A}(P)$. Dafür muss $\mathcal{A}(P) = U$ sein.
Nach Definition 9.6 ist aber das Universum einer Struktur nicht leer. Also muss für alle Strukturen $\mathcal{A}(\forall x\ P(x) \Rightarrow \exists y\ P(y)) = \mathbf{1}$ gelten.

Semantik der Prädikatenlogik
Das Problem der systematischen Berücksichtigung aller Strukturen

Auswertung (einer Formel) in der Aussagenlogik

- $\mathcal{L}_{AL}$ enthält nur Formeln mit endlicher Länge,
  d.h. jede Formel enthält nur eine endliche Anzahl von atomaren Formeln.
- Die Bewertung erfolgt über der Menge der Wahrheitswerte $\{\mathbf{0}, \mathbf{1}\}$
- Es gibt nur endlich viele zu unterscheidende Belegungen $\mathcal{A}_i$ für eine Formel $F \in \mathcal{L}_{AL}$.

Auswertung (einer Formel) in der Prädikatenlogik

- Da es für eine Formel $F \in \mathcal{L}_{PL}$ unendlich viele Strukturen $\mathcal{A}$ gibt,
  ist eine systematische, erschöpfende Berechnung aller Auswertungen der Formel F grundsätzlich nicht möglich.
- Es werden andere Techniken benötigt, um semantische Eigenschaften zu prüfen.

Erfüllbarkeit, Folgerung: Einige wichtige Theoreme

Die folgenden Sätze gelten entsprechend zur Aussagenlogik:

Satz 9.14: Eine Formel F ist genau dann gültig, wenn $\neg F$ unerfüllbar ist.
Satz 9.15: $F \vDash G$ genau dann, wenn $F \wedge \neg G$ unerfüllbar ist.

- Durch Satz 9.14 wird die Gültigkeit von Formeln auf die Unerfüllbarkeit von Formeln zurückgeführt.
- Durch Satz 9.15 wird Folgerbarkeit auf Unerfüllbarkeit zurückgeführt.

- Entsprechend der Aussagenlogik werden mechanische Verfahren zur Prüfung der Unerfüllbarkeit untersucht.

  ➔ Deduktions- und Widerlegungsverfahren.

Prädikatenlogik → Aussagenlogik: Semantische Perspektive

Reduktion der Prädikatenlogik auf die Aussagenlogik:

- Wenn kein Prädikatensymbol (mit Stelligkeit > 0) in einer prädikatenlogischen Sprache vorhanden ist, so sind alle atomaren Formeln Aussagensymbole.
  - Terme, Variablen und Funktionssymbole, aber insbesondere auch Quantoren sind in einer derartigen prädikatenlogischen Sprache überflüssig.
  - Die prädikatenlogische Semantik wird zur aussagenlogischen Semantik reduziert.
- Wenn in einer prädikatenlogischen Sprache weder Variablen noch Quantoren auftreten, ist ebenfalls eine Reduktion auf die Aussagenlogik möglich.
  - Die atomaren Formeln einer derartigen prädikatenlogischen Sprache können als atomare Formeln einer aussagenlogischen Sprache angesehen werden.
  - Auch in diesem Fall wird die prädikatenlogische Semantik zur aussagenlogischen Semantik reduziert.
  - ABER: Die zusätzliche Ausdrucksstärke der Prädikatenlogik geht dabei verloren!

Prädikatenlogik 2. Stufe

Prädikatenlogik 1. Stufe

- Im bisher vorgestellten Typ der Prädikatenlogik (Prädikatenlogik 1. Stufe) gibt es:
  - eine Art von Variablen, nämlich *Individuenvariablen*, die Terme sind,
  - eine Art von Quantoren, nämlich solche, die Individuenvariablen binden.

➔ Semantik für Formeln mit Quantoren (x-Varianten)

Jenseits der Prädikatenlogik 1. Stufe

- Erweiterung zu Quantifizierungen über Funktionen, Eigenschaften und Relationen
  - zusätzliche Arten von Variablen: Funktions-, Eigenschafts- und Relationsvariablen, denen eine eindeutige Stelligkeit zugewiesen ist.
  - die Quantoren, binden Variablen aller Typen.

➔ Erweiterung der Semantik für Formeln mit Quantoren ist notwendig.

- Vergrößerung der Ausdrucksstärke:
  - Peanos Axiomatisierung der Natürlichen Zahlen

$\forall P\ ((P(1) \wedge \forall n\ (P(n) \Rightarrow P(n+1))) \Rightarrow \forall n\ P(n))$ [„Induktionsaxiom"]

Zum Selbststudium: Funktionssymbole

Wann nutzt man Funktionssymbole, wann Relationssymbole

- Funktionssymbole in der Logiksprache sind im wesentlichen an zwei Stellen nützlich
  - Formalisierung der Mathematik
  - Zur Definition von Verarbeitungstechniken und Beweisen über die Logik (Wir werden später sog. Skolemfunktionssybole nutzen und damit gewisse Existenzquantoren überflüssig machen.)
- Die Übersetzung natürlichsprachlicher Ausdrücke in die Logik erfordert eigentlich nie die Nutzung von Funktionssymbolen.
- Bei der einfachen Logik, die wir hier nutzen, sind Funktionssybole auch etwas tückisch, da man sie mit jedem Term kombinieren kann und dann ein neuer Term entsteht, der wieder etwas bezeichnet. Das geht in der Mathematik für arithmetische Operationen ganz gut, aber schon die formale Erfassung der Tatsache, dass man nichts durch 0 teilen darf, ist eine echte Herausforderung der logischen Modellierung, mit der wir uns hier lieber nicht befassen wollen.

Wichtige Konzepte dieser Vorlesung

Syntax der Prädikatenlogik

- Variable, Quantor, Konstante, Funktionssymbol, Prädikatensymbol / Relationssymbol, Term, Stelligkeit
- Teilterm, Teilformel, Strukturbaum
- freie Variable (in einer Formel), durch Quantor gebundene Variable (in einer Position), Quantorenvariable, Skopus eines Quantors, geschlossene Formel

Semantik der Prädikatenlogik

- Struktur, Universum / Domäne, Interpretation / Auswertung
- Bestimmung des Wertes einer Formel durch eine Struktur / Auswertung einer Formel
- x-Variante einer Struktur
- Modell, erfüllbar, falsifizierbar, unerfüllbar (allgemein-)gültig
- Äquivalenz, Folgerung