Formale System short summary

Adam Urban
16. Februar 2006
made with $\LaTeX\,2_\varepsilon$

Inhaltsverzeichnis

1 Aussagenlogik **4**
1.1 Wichtige Definitionen, Lemmas und Sätze 4
1.2 Beweistheorie 6

2 Prädikatenlogik erster Ordnung **9**
2.1 Wichtige Definitionen, Lemmas und Sätze 9
2.2 Normalformen 15
2.3 Beweistheorie 17
2.4 Weitere Anmerkungen zur Prädikatenlogik erster Ordnung 21
2.5 Anwendung 22

3 Die Behandlung der Gleichheit **22**
3.1 Abstrakte Datentypen 22
3.2 Birkhoffs Kalkül 24
3.3 Reduktionssysteme 25
3.4 Termersetzungssysteme 26

4 Stärkere Logiken **26**
4.1 Prädikatenlogik zweiter Ordnung 26
4.2 Dynamische Logik 28

5 Omega Automaten **30**

6 Temporale Logiken **33**
6.1 Lineare Temporale Logik 33

1 Aussagenlogik

1.1 Wichtige Definitionen, Lemmas und Sätze

- Es sei Σ eine aussagenlogische Signatur. Eine **Interpretation** über Σ ist eine beliebige Abbildung

  $$I : \Sigma \to \{W, F\}.$$

  Zu jeder Interpretation I über Σ wird nun die zugehörige Auswertung der Formeln über Σ definiert als die Abbildung

  $$val_I : For0_\Sigma \to \{W, F\}$$

  mit:

  $$\begin{array}{rcll} val_I(1) & = & W & \\ val_I(0) & = & F & \\ val_I(P) & = & I(P) & \quad \text{für jedes } P \in \Sigma \end{array}$$

  $\looparrowright$ [Definition 2.10, Skript S. 11]

- Ein **Modell** einer Formel $A \in For0_\Sigma$ ist eine Interpretation I über Σ mit $val_I(A) = W$. Zu einer Formelmenge $M \subseteq For0_\Sigma$ ist ein Modell von M eine Interpretation I, welche Modell von jedem $A \in M$ ist.
  $\looparrowright$ [Definition 2.11, Skript S. 11]

- $A \in For0_\Sigma$ heisst **allgemeingültig** (über Σ) $:\Leftrightarrow$ $val_I(A) = W$ für jede Interpretation I über Σ. A heisst **erfüllbar** (über Σ) $:\Leftrightarrow$ es gibt eine Interpretation I über Σ mit $val_I(A) = W$.
  $\looparrowright$ [Definition 2.14, Skript S. 13]

- Es gilt

A ist erfüllbar	$\Leftrightarrow$	$\neg A$ ist nicht allgemeingültig
A ist allgemeingültig	$\Leftrightarrow$	$\neg A$ ist nicht erfüllbar

  $\looparrowright$ [Korollar 2.15, Skript S. 13]

- Es seien: Σ eine Signatur, $M \subseteq For0_\Sigma$, $A \in For0_\Sigma$. Wir definieren

  $$M \models A \quad \text{gdw} \quad \text{Jedes Modell von } M \text{ ist auch Modell von } A$$

  $M \models_\Sigma A$ wird gelesen als: aus M folgt A
  $\looparrowright$ [Definition 2.16, Skript S. 15]

- Es gilt

  - $\models A \Leftrightarrow A$ ist allgemeingültig
  - $\models \neg A \Leftrightarrow A$ ist unerfüllbar
  - $A \models B \Leftrightarrow A \to B$
  - $M \cup \{A\} \models B \Leftrightarrow M \models A \to B$

  $\looparrowright$ [Korollar 2.17, Skript S. 15]

- A, B heissen **logisch äquivalent** $:\Leftrightarrow$ $A \models_\Sigma B$ und $B \models_\Sigma A$
  $\looparrowright$ [Definition 2.18, Skript S. 15]

- Logische Äquivalenz ist bezüglich der aussagenlogischen Operatoren eine Kongruenzrelation auf $For0_\Sigma$. Insbesondere gilt für beliebige $A \in For0_\Sigma$

A ist allgemeingültig	$\Leftrightarrow$	A ist logisch äquivalent zu 1
A ist unerfüllbar	$\Leftrightarrow$	A ist logisch äquivalent zu 0

  $\looparrowright$ [Korollar 2.21, Skript S. 16]

- Wir betrachten aussagenlogische Formeln, die aufgebaut sind aus dem dreistelligen Operator sh, den Konstanten 0, 1 und Aussagevariablen $P_1, \ldots, P_n, \ldots$. Wir wollen solche Formeln ***sh*-Formeln** nennen und die Menge aller sh-Formeln mit $shFor0$ bezeichnen. Der Wahrheitswerteverlauf von sh (sh als Referenz auf Shannon) wird gegeben durch

$$sh(P_1, P_2, P_3) = \begin{cases} P_2 & \text{falls} \quad P_1 = F \\ P_3 & \text{falls} \quad P_1 = W \end{cases}$$

  ↬ [Definition 2.33, Skript S. 23]

- Wir fixieren eine Ordnung auf der Menge der Aussagevariablen, etwa die durch die Ordnung der Indizes gegebene. Eine sh-Formel heisst eine **normierte *sh*-Formel**, wenn sie der folgenden rekursiven Definition genügt.

  1. Die Konstanten 0, 1 sind normierte sh-Formeln
  2. Sei P_i eine Aussagevariable, seien A, B normierte sh-Formeln, so dass keine Aussagevariable P_j mit $i \geq j$ in A oder B auftritt. Dann ist $sh(P_i, A, B)$ eine normierte sh-Formel.

  ↬ [Definition 2.35, Skript S. 25]

- Ein sh-Graph heisst **reduziert** (**BDD**), wenn

  1. es keine zwei Konoten v und w ($v \neq w$) gibt, so dass der in v verwurzelte Teilgraph G_v mit dem in w verwurzelten Teilgraph G_w isomorph ist.
  2. es keinen Knoten v gibt, so dass beide von v ausgehenden Kanten zum selben Nachfolgerknoten führen.

  ↬ [Definition 2.40, Skript S. 28]

- Eine **Krom-Formel** ist eine aussagenlogische Formel in KNF, in der jede Disjunktion höchstens zwei Literale enthält.
  Für Krom-Formeln ist die Erfüllbarkeit in polynomialer Zeit entscheidbar.
  ↬ [Definition 2.60, Skript S. 34]

- Eine **Horn-Formel** ist eine aussagenlogische Formel in KNF, in der jede Disjunktion höchstens ein positives Literal enthält. Eine solche Disjunktion heisst Horn-Klausel. Für Horn-Formeln ist die Erfüllbarkeit in quadratischer Zeit entscheidbar.
  Horn-Klauseln schreibt man häufig logisch äquivalent als Implikation:

$\neg B_1 \vee \ldots \vee \neg B_m \vee A$	$\neg B_1 \vee \ldots \vee \neg B_m \rightarrow A$
$\neg B_1 \vee \ldots \vee \neg B_m$	$\neg B_1 \vee \ldots \vee \neg B_m \rightarrow 0$
A	A

  ↬ [Definition 2.62, Skript S. 34]

- Eine Formel A aus $AeqFor$ ist eine **Tautologie**
  gdw.
  jede Aussagenvariable und die Konstante 0 hat eine gerade Anzahl von Vorkommen in A.
  ↬ [Satz 2.65, Skript S. 37]

- Eine **definite Horn-Formel** ist eine aussagenlogische Formel in KNF, in der jede Disjunktion genau ein positives Literal enthält. Eine solche Disjunktion heisst definite Horn-Klausel. Definite Horn-Formeln sind stets erüllbar.
  ↬ [Definition 2.67, Skript S. 40]

1.2 Beweistheorie

- In einer Sprache $(\triangle, L)$ sei ***Kal*** ein **Kalkül** und es sei M eine Teilmenge von L. Eine **Ableitung** aus M in Kal ist eine Folge

$$(u_1, \ldots, u_m)$$

von Wörtern in L, so dass für jedes $i \in \{1, \ldots, m\}$ gilt:

  - u_i ist Axiom; oder
  - $u_i \in M$; oder
  - es gibt eine Regel $R \in Kal$ einer Stelligkeit $n \geq 1$ sowie Indizes $j_1, \ldots, j_n \in \{1, \ldots, i-1\}$, so dass $(u_{j_1}, \ldots, u_{j_n}, u_i\} \in R$.

  Im letzten Fall entsteht u_i durch Anwendung von R auf gewisse vorangegangene u_j. Man beachte, dass für $i = 1$ gilt $\{1, \ldots, i-1\} = \emptyset$, d.h. u_1 muss Axiom oder ein Element von M sein.
  $u \in L$ heisst **ableitbar** aus M in Kal, kurz

$$M \vdash_{Kal} u$$

genau dann, wenn es eine Ableitung $(u_1, \ldots, u_m)$ in Kal gibt mit $u_m = u$. Für $\emptyset \vdash_{Kal} u$ schreiben wir $\vdash_{Kal} u$, für $\{v\} \vdash_{Kal} u$ schreiben wir $v \vdash_{Kal} u$.
↬ [Definition 3.2, Skript S. 42]

- Die **aussagenlogische Resolution** (über Σ) ist die Regel

$$\frac{C_1 \cup \{P\}, C_2 \cup \{\neg P\}}{C_1 \cup C_2} \quad \text{mit } P \in \Sigma, C_1, C_2 \text{ Klauseln über } \Sigma$$

$C_1 \cup C_2$ heisst Resolvente von $C_1 \cup \{P\}, C_2 \cup \{\neg P\}$.
↬ [Definition 3.19, Skript S. 53]

- Eine **Vorzeichenformel** ist eine Zeichenkette der Gestalt $0A$ oder $1A$ mit $A \in For0_\Sigma$. Dabei sind 0, 1 neue Sonderzeichen, die Vorzeichen, im Alphabet der Objektsprache.
  Vorzeichen treten nur einmal am Beginn einer Formel auf, deswegen ist auch eine Verwechslung mit den Booleschen Konstanten ausgeschlossen.
  ↬ [Definition 3.26, Skript S. 58]

- Es gibt folgende **Typen der Vorzeichenformeln**:

Typ ϵ	(elementarer Typ):	$0P$, $1P$ für $P \in \Sigma$
Typ α	(konjunktiver Typ):	$0\neg B$, $1\neg B$, $1(B \land C)$, $0(B \lor C)$, $0(B \to C)$
Typ β	(disjunktiver Typ):	$0(B \land C)$, $1(B \lor C)$, $1(B \to C)$

↬ [Definition 3.28, Skript S. 58]

- Zu jeder nicht elementaren Vorzeichenformel V definieren wir zwei **Abkömmlinge** V_1, V_2 (die auch identisch sein könnten) gemäß den folgenden Tabellen

vom Typ α	V_1	V_2
$0\neg B$	$1B$	$1B$
$1\neg B$	$0B$	$0B$
$1(B \land C)$	$1B$	$1C$
$0(B \lor C)$	$0B$	$0C$
$0(B \to C)$	$1B$	$0C$

vom Typ β	V_1	V_2
$0(B \land C)$	$0B$	$0C$
$1(B \lor C)$	$1B$	$1C$
$1(B \to C)$	$0B$	$1C$

Elementare Vorzeichenformeln haben keine Abkömmlinge.
↬ [Definition 3.28, Skript S. 58]

- Es sei M eine Menge von Formeln und A eine Formel. Ein (aussagenlogisches) **Tableau** für A über M ist ein Baum mit Beschriftung der Knoten durch markierte Formeln, gemäß der folgenden rekursiven Definition:

1. Ein einziger mit $0A$ markierter Knoten ist ein Tableau für A über M.

2. Es sei T ein Tableau für A über M und π ein Pfad in T, so dass auf π ein Knoten liegt, der mit einem V vom Typ α beschriftet ist. Dann ist auch T_1 ein Tableau für A über M, welches dadurch aus T entsteht, dass an π zwei neue Knoten angefügt werden, welche mit V_1 und mit V_2 beschriftet sind (Abb. 3.1, Skript S. 60). Wir sagen in diesem Fall, dass T_1 aus T durch eine Anwendung der **α-Regel** aus T hervorgeht.

3. Es sei T ein Tableau für A über M und π ein Pfad in T, so dass auf π ein Knoten liegt, der mit einem V vom Typ β beschriftet ist. Dann ist auch T_2 ein Tableau für A über M, welches dadurch entsteht, dass an π zwei verzweigende Kanten angefügt werden, welche zu neuen Blättern mit Beschriftungen V_1 bzw. V_2 führen (Abb. 3.1, Skript S. 60). Wir sagen, T_2 geht aus T durch eine Anwendung der **β-Regel** aus T hervor.

4. Es sei T ein Tableau für A über M und π ein Pfad in T, dann ist auch T_3 ein Tableau für A über M, welches durch Verlängerung von π um einen mit $1B$ beschrifteten Knoten entsteht, für eine Formel B aus M. Wir sagen, T_3 geht aus T durch eine Anwendung der **Voraussetzungsregel**.

↬ [Definition 3.31, Skript S. 60]

- Ein Tableau T heisst **geschlossen**, wenn alle seine Pfade geschlossen sind. Ein Pfad π heisst geschlossen, wenn zwei Knoten auf π liegen, welche respektive beschriftet sind mit $0A$ und $1A$, für eine beliebige Formel A. Gelegentlich deuten wir die Abgeschlossenheit eine Pfades π an durch Anfügen einer Kante an π, welche zu einem mit * beschrifteten Blatt führt.
  Wir benutzen die folgende Kurzschreibweise für Tableauregeln

$$\begin{array}{c} \alpha \\ \hline \alpha_1 \\ \alpha_2 \end{array} \qquad \begin{array}{c|c} \multicolumn{2}{c}{\beta} \\ \hline \beta_1 & \beta_2 \end{array}$$

↬ [Definition 3.32, Skript S. 61]

- Es sei $A \in For0_\Sigma$ und $M \subseteq For0_\Sigma$. A ist aus M ableitbar in **T0**,

$$M \vdash_{T0} A,$$

  gdw. es ein geschlossenes Tableau für $0A$ über M gibt.
  ↬ [Definition 3.33, Skript S. 62]

- Das Tableau T für A über der endlichen Formelmenge E heisst **erschöpft**, wenn auf jedem offenen Pfad π von T jede α- und β-Formel benutzt wurde (d.h. die entsprechende Regel auf sie angewandt wurde), und jedes $1B$, $B \in E$, auf π vorkommt.
  ↬ [Definition 3.38, Skript S. 63]

- Eine **Sequenz** wird notiert als eine Folge zweier endlicher Mengen aussagenlogischer Formeln, getrennt durch das Symbol $\Rightarrow$:

$$\Gamma \Rightarrow \Delta.$$

  Γ wird **Antezedent** und Δ **Sukzedent** genannt. Sowohl links wie rechts vom Sequenzenpfeil $\Rightarrow$ kann auch die leere Folge stehen. Wir schreiben dann

$$\Rightarrow \Delta \quad \text{bzw.} \quad \Gamma \Rightarrow \quad \text{bzw.} \quad \Rightarrow .$$

↬ [Definition 3.48, Skript S. 68]

- **Axiome und Regeln** des Sequenzenkalküls **S0**:

  axiom

  $$\frac{}{\Gamma, F \Rightarrow F, \Delta}$$

  not-left

  $$\frac{\Gamma, \Rightarrow F, \Delta}{\Gamma, \neg F \Rightarrow \Delta}$$

  and-left

  $$\frac{\Gamma, F, G \Rightarrow \Delta}{\Gamma, F \wedge G \Rightarrow \Delta}$$

  not-right

  $$\frac{\Gamma, F \Rightarrow \Delta}{\Gamma \Rightarrow \neg F, \Delta}$$

  and-right

  $$\frac{\Gamma \Rightarrow F, \Delta \quad \Gamma \Rightarrow G, \Delta}{\Gamma \Rightarrow F \wedge G, \Delta}$$

  impl-left

  $$\frac{\Gamma \Rightarrow F, \Delta \quad \Gamma, G \Rightarrow \Delta}{\Gamma, F \rightarrow G \Rightarrow \Delta}$$

  or-left

  $$\frac{\Gamma, F \Rightarrow \Delta \quad \Gamma, G \Rightarrow \Delta}{\Gamma, F \vee G \Rightarrow \Delta}$$

  impl-right

  $$\frac{\Gamma, F \Rightarrow G, \Delta}{\Gamma \Rightarrow F \rightarrow G, \Delta}$$

  or-right

  $$\frac{\Gamma \Rightarrow F, G, \Delta}{\Gamma \Rightarrow F \vee G, \Delta}$$

  $\looparrowright$ [Definition 3.50, Skript S. 68]

- Für $A \in For0_\Sigma$, $M \subseteq For0_\Sigma$ sagen wir, dass A aus M ableitbar in **S0** ist, in Zeichen:

  $$M \vdash_S A,$$

  genau dann, wenn ein geschlossener Beweisbaum mit Wurzelmarkierung $M \Rightarrow A$ in **S0** existiert.

  $\looparrowright$ [Definition 3.52, Skript S. 69]

- Die **1-Resolution** (**unit resolution**) benutzt dieselbe Notation wie im Resolutionskalkül. Die Resolutionsregel ist ein Spezialfall der allgemeinen Resolutionsregel:

  $$\frac{\{P\}, C_2 \cup \{\neg P\}}{C_2} \qquad \frac{\{\neg P\}, C_2 \cup \{P\}}{C_2}$$

  Die 1-Resolutionsregel ist nicht vollständig.

  $\looparrowright$ [Skript S. 73]

- **Davis-Putnam-Loveland Verfahren**:

  1. Programm widerlege(S);
  2. Falls $S = \emptyset$, Ende (S ist erfüllbar).
  3. Falls S keine Einerklausel enthält, wähle eine Variable P;
     widerlege(S_P); widerlege($S_{\neg P}$);
  4. Sonst wähle eine Einerklausel $K \in S$;
  5. S = reduziere(K, S)
     - Lasse alle Klauseln weg, die K als Teilklausel enthalten,
     - Lasse in allen übrigen Klauseln das zu K komplementäre Literal weg.
  6. Falls $\Box \in S$, Ende (S widersprüchlich),
     sonst widerlege(S);

  $\looparrowright$ [Skript S. 73]

- **Numerische Verfahren**:
  Gegeben sei eine KNF $A = D_1 \wedge \ldots \wedge D_k$. U_i entstehe aus D_i, indem P_j ersetzt wird durch X_j, $\neg P_j$ durch $(1 - X_j)$ und $\wedge$ durch $+$. $U(A)$ ist die Menge der Ungleichungen

  $$U_i \geq 1 \quad \text{für alle } i$$

  und

  $$0 \leq X_j \leq 1 \quad \text{für alle } j$$

  A ist erfüllbar gdw $U(A)$ in den ganzen Zahlen lösbar ist.

  $\looparrowright$ [Satz 3.58, Skript S. 74]

2 Prädikatenlogik erster Ordnung

2.1 Wichtige Definitionen, Lemmas und Sätze

- Die folgenden Zeichen sind in jeder Sprache der **PL1** vorhanden. Sie heissen **logische Zeichen** (manchmal auch Sonderzeichen).
  wie in der Aussagenlogik:

  $$(\quad , \quad) \quad , \quad 1 \quad , \quad 0 \quad , \quad \neg \quad , \quad \wedge \quad , \quad \vee \quad , \quad \rightarrow \quad , \quad \leftrightarrow$$

  neu:

  , (Komma) , $\doteq$ (objektsprachliches Gleichheitssymbol) , $\forall$ (Allquantor)
  $\exists$ (Existenzquantor) , v_i (Individuenvariablen, $i \in \mathbb{N}$)

  $Var := \{v_i \,|\, i \in \mathbb{N}\}$ ist die Menge der **Individuenvariablen** oder kurz **Variablen**. Var ist disjunkt zur Menge der übrigen Sondersymbole.
  $\looparrowright$ [Definition 4.1, Skript S. 84]

- Eine **Signatur** der **PL1** ist ein Tripel $\Sigma = (F_\Sigma, P_\Sigma, \alpha_\Sigma)$ mit

  - F_Σ, P_Σ sind endliche oder abzählbar unendliche, effektiv gegebene Mengen
  - F_Σ, P_Σ und die Menge der Sondersymbole sind paarweise disjunkt
  - $\alpha_\Sigma : F_\Sigma \cup P_\Sigma \rightarrow \mathbb{N}$.

  Die $f \in F_\Sigma$ heissen **Funktionssymbole**, die $p \in P_\Sigma$ **Prädikatssymbole**, α_Σ ordnet jedem Funktions- oder Prädikatssymbol seine **Stelligkeit** zu: f ist n-stelliges Funktionssymbol, wenn $\alpha_\Sigma(f) = n$; entsprechend für $p \in P_\Sigma$.
  Ein nullstelliges Funktionssymbol heisst auch **Konstantensymbol** oder kurz **Konstante**, ein nullstelliges Prädikatssymbol ist ein **aussagenlogisches Atom**.
  $\looparrowright$ [Definition 4.2, Skript S. 84]

- $Term_\Sigma$, die Menge der **Terme** über Σ, ist induktiv definiert durch

  1. $Var \subseteq Term_\Sigma$
  2. Mit $f \in F_\Sigma$, $\alpha_\Sigma(f) = n$, $t_1, \ldots, t_n \in Term_\Sigma$ ist auch $f(t_1, \ldots, t_n) \in Term_\Sigma$

  Man beachte, dass jede Konstante ein Term ist. Ein Term heisst **Grundterm**, wenn er keine Variablen enthält.
  $\looparrowright$ [Definition 4.3, Skript S. 85]

- At_Σ, die Menge der **atomaren Formeln** – oder **Atome** – über Σ, ist definiert als

  $$At_\Sigma := \{s \doteq t \,|\, s, t \in Term_\Sigma\} \cup \{p(t_1, \ldots, t_n)\} \,|\, p \in P_\Sigma,\ \alpha_\Sigma(p) = n,\ t_1, \ldots, t_n \in Term_\Sigma\}$$

  $\looparrowright$ [Definition 4.4, Skript S. 85]

- For_Σ, die Menge der **Formeln** über Σ, ist induktiv definiert durch

  1. $\{1, 0\} \cup At_\Sigma \subseteq For_\Sigma$
  2. Mit $x \in Var$ und $A, B \in For_\Sigma$ sind ebenfalls in For_Σ:

  $$\neg A \quad , \quad (A \wedge B) \quad , \quad (A \vee B) \quad , \quad (A \rightarrow B) \quad , \quad (A \leftrightarrow B) \quad , \quad \forall x A \quad , \quad \exists x A$$

  $\looparrowright$ [Definition 4.5, Skript S. 85]

- Ein **Teilterm** oder **Unterterm** eines Terms t ist ein Teilwort von t, das selbst Term ist; entsprechend sind die Teilterme (Unterterme) einer Formel definiert. Eine Teilformel einer Formel A ist ein Teilwort von A, das selbst eine Formel ist.
  $\looparrowright$ [Definition 4.6, Skript S. 86]

- Hat eine Formel A die Gestalt $\forall xB$ oder $\exists xB$ (für ein $x \in Var$ und $B \in For_\Sigma$), so heisst B der **Wirkungsbereich** des Präfixes $\forall x$ bzw. $\exists x$ von A.
  $\looparrowright$ [Definition 4.8, Skript S. 86]

- Ein Auftreten einer Variablen x in einer Formel A heisst **gebunden**, wenn es innerhalb des Wirkungsbereichs eines Präfixes $\forall x$ bzw. $\exists x$ einer Teilformel von A stattfindet.
  $\looparrowright$ [Definition 4.9, Skript S. 86]

- Ein Auftreten einer Variablen x in einer Formel A heisst **frei**, wenn es nicht gebunden ist und nicht unmittelbar rechts neben einem Quantor stattfindet.
  $\looparrowright$ [Definition 4.10, Skript S. 86]

- Es sei $A \in For_\Sigma$ und $t \in Term_\Sigma$.

$$\begin{aligned} Bd(A) &:= \{x \mid x \in Var,\ x \text{ tritt gebunden in } A \text{ auf}\} \\ Frei(A) &:= \{x \mid x \in Var,\ x \text{ tritt frei in } A \text{ auf}\} \\ Var(A) &:= Frei(A) \cup Bd(A) \\ Var(t) &:= \{x \mid x \in Var,\ x \text{ kommt in } t \text{ vor}\} \end{aligned}$$

  $t \in Term_\Sigma$ heisst Grundterm, wenn $Var(t) = \{\}$.
  $\looparrowright$ [Definition 4.12, Skript S. 87]

- A heisst **geschlossen**, wenn $Frei(A) = \{\}$. Ist allgemein $Frei(A) = \{x_1, \ldots, x_n\}$, so heisst

  $\forall x_1 \ldots \forall x_n A$ **Allabschluss** und $\exists x_1 \ldots \exists x_n A$ **Existenzabschluss**

  von A. Abkürzend schreiben wir $Cl_\forall A$ bzw. $Cl_\exists A$. Ist A geschlossen, dann gilt also $Cl_\forall A = Cl_\exists A = A$.
  $\looparrowright$ [Definition 4.13, Skript S. 87]

- Eine **Substitution** (über Σ) ist eine Abbildung

$$\sigma : Var \to Term_\Sigma$$

  mit $\sigma(x) = x$ für fast alle $x \in Var$. Sind $x_1, \ldots, x_m$ so, dass gilt $\{x \mid \sigma(x) \neq x\} \subseteq \{x_1, \ldots, x_m\}$, und ist $\sigma(x_i) = s_i$ für $i = 1, \ldots, m$, so geben wir σ auch an in der Schreibweise

$$\{x_1/s_1, \ldots, x_m/s_m\}.$$

  σ heisst **Grundsubstitution**, wenn für alle $x \in Var$ gilt: $\sigma(x) = x$ oder $\sigma(x)$ ist Grundterm.
  $\looparrowright$ [Definition 4.14, Skript S. 87]

- Wir setzen eine Substitution σ fort zu Abbildungen

$$\begin{gathered} Term_\Sigma \to Term_\Sigma \\ For_\Sigma \to For_\Sigma \end{gathered}$$

  die beide wieder mit σ bezeichnet werden, mittels:

  - $\sigma(f(t_1, \ldots, t_n)) = f(\sigma(t_1), \ldots, \sigma(t_n))$
  - $\sigma(p(t_1, \ldots, t_n)) = p(\sigma(t_1), \ldots, \sigma(t_n))$
  - $\sigma(t \doteq s) = \sigma(t) \doteq \sigma(s)$
  - $\sigma(\neg A) = \neg\sigma(A)$
  - $\sigma(A \circ B) = \sigma(A) \circ \sigma(B)$ für jeden zweistelligen aussagenlogischen Operator $\circ$.
  - $\sigma(QxA) = Qx\sigma_x(A)$, wobei $\sigma_x(x) = x$ und $\sigma_x(y) = \sigma(y)$ für $y \neq x$, für $Q \in \{\forall, \exists\}$

  $\sigma(A)$ entsteht aus A, indem simultan für jedes $x \in Var$ an jeder Stelle, wo x frei in A auftritt, x ersetzt wird durch $\sigma(x)$.
  $\looparrowright$ [Definition 4.15, Skript S. 88]

- Eine Substitution σ heisst **kollisionsfrei** für eine Formel A, wenn für jede Variable z und jede Stelle freien Auftretens von z in A gilt: Diese Stelle liegt nicht im Wirkungsbereich eines Präfixes $\forall x$ oder $\exists x$, wo x eine Variable in $\sigma(z)$ ist.
  $\looparrowright$ [Definition 4.17, Skript S. 88]

- Sind σ, τ Substitutionen, dann definieren wir die **Komposition** von τ und σ durch

$$(\tau \circ \sigma)(x) = \tau(\sigma(x)).$$

  $\looparrowright$ [Definition 4.18, Skript S. 89]

- Eine **Variablenumbenennung** ist eine Substitution σ mit

$$\sigma(x) \in Var \text{ für alle } x \in Var \qquad \text{und} \qquad \sigma \text{ ist injektiv}$$

  $\looparrowright$ [Definition 4.20, Skript S. 89]

- Gilt für Substitutionen σ, τ, dass $\sigma \circ \tau = id$, dann ist σ eine Variablenumbenennung.
  $\looparrowright$ [Korollar 4.21, Skript S. 89]

- Es sei $T \subseteq Term_\Sigma$, $T \neq \{\}$ und σ eine Substitution über Σ. σ **unifiziert** T, oder: ist **Unifikator** von T, genau dann, wenn $\#\sigma(T) = 1$. T heisst **unifizierbar**, wenn T einen Unifikator besitzt. Insbesondere sagen wir für zwei Terme s, t dass s unifizierbar sei mit t, wenn $\{s, t\}$ unifizierbar ist, d.h. in diesem Fall, wenn $\sigma(t) = \sigma(s)$.
  Die Begriffe werden auf Formelmengen übertragen durch: σ unifiziert M genau dann, wenn $\#\sigma(M) = 1$.
  $\looparrowright$ [Definition 4.22, Skript S. 90]

- Es sei $T \subseteq Term_\Sigma$. Ein **allgemeinster Unifikator** oder **mgu** (most general unifier) von T ist eine Substitution μ mit

  1. μ unifiziert M
  2. Zu jedem Unifikator σ von T gibt es eine Substitution σ' mit $\sigma = \sigma' \circ \mu$.

  $\looparrowright$ [Definition 4.26, Skript S. 91]

- Es sei T eine nichtleere Menge von Termen. Dann ist jeder allgemeinste Unifikator von T bis auf Variablenumbenennung eindeutig bestimmt, d.h.: Sind μ, μ' allgemeinste Unifikatoren von T mit $\mu(T) = \{t\}$ und $\mu'(T) = \{t'\}$, dann gibt es eine Variablenumbenennung π mit $t' = \pi(t)$.
  $\looparrowright$ [Lemma 4.27, Skript S. 91]

- **Unifikationsalgorithmus**
  Eingabe: eine nicht-leere Literalmenge L

  sub := []; (dies ist die leere Substitution)
  while $|L\,sub| > 1$ **do**
      begin
          Durchsuche die Literale in $L\,sub$ von links nach rechts, bis die erste Position gefunden ist, wo sich mindestens zwei Literale (sagen wir L_1 und L_2) in den vorkommenden Zeichen unterscheiden;
          if keines der beiden Zeichen ist eine Variable
              then
                  stoppe mit der Ausgabe "nicht unifizierbar"
              else
                  begin
                      Sei x die Variable und t sei der im anderen Literal beginnende Term (dies kann auch eine Variable sein);
                      if x kommt in t vor
                          then
                              stoppe mit der Ausgabe "nicht unifizierbar"
                          else
                              $sub := sub[x/t]$;
                              (dies bedeutet die Hintereinanderausführung von sub und $[x/t]$)
                  end;
      end;

  ↬ [Schöning S. 90]
  ↬ [Skript S. 92]

- Es sei Σ eine Signatur der **PL1**. Eine **Interpretation** D von Σ ist ein Paar (D, I) mit

  1. D ist eine beliebige, nichtleere Menge
  2. I ist eine Abbildung der Signatursymbole, die
     - jeder Konstanten c ein Element $I(c) \in D$
     - für $n \geq 1$: jedem n-stelligen Funktionssymbol f eine Funktion $I(f) : D^n \to D$
     - jedem 0-stelligen Prädikatsymbol P eine Wahrheitswert $I(P) \in \{W, F\}$
     - für $n \geq 1$: jedem n-stelligen Prädikatsymbol p eine n-stellige Relation $I(p) \subseteq D^n$

     zuordnet.

  ↬ [Definition 4.35, Skript S. 97] Es sei (D, I) eine Interpretation von Σ. Eine **Variablenbelegung** (oder kurz Belegung über D) ist eine Funktion

  $$\beta : Var \to D.$$

  Zu $\beta, x \in Var$ und $d \in D$ definieren wir die **Modifikation** von β an der Stelle x zu d:

  $$\beta_x^d(y) = \begin{cases} d & \text{falls} \quad y = x \\ \beta(y) & \text{falls} \quad y \neq x \end{cases}$$

  ↬ [Definition 4.36, Skript S. 98]

- Es gelten die aus der Aussagenlogik bekannten Abhängigkeiten zwischen den Bedeutungen der Operatoren *true*, *false*, $\neg$, $\wedge$, $\vee$, $\to$, $\leftrightarrow$, so dass hier wieder die bekannten Teilmengen dieser Mengen ausreichen. Ferner gilt für alle x, A:

  $$\begin{aligned} val_{D,I,\beta}(\exists x A) &= val_{D,I,\beta}(\neg\forall x \neg A) \\ val_{D,I,\beta}(\forall x A) &= val_{D,I,\beta}(\neg\exists x \neg A) \end{aligned}$$

  d.h. $\exists$ ist durch $\forall$ und $\neg$, $\forall$ ist durch $\exists$ und $\neg$ ausdrückbar.
  ↬ [Korollar 4.39, Skript S. 100]

- D sei Interpretation über Σ.

  1. Gilt für den Term t und die Variablenbelegung β, γ, dass
  $$\beta(x) = \gamma(x) \text{ für alle } x \in Var(t),$$
  dann
  $$val_{D,\beta}(t) = val_{D,\gamma}(t).$$
  2. Gilt für die Formel A und die Variablenbelegung β, γ, dass
  $$\beta(x) = \gamma(x) \text{ für alle } x \in Frei(A),$$
  dann
  $$val_{D,\beta}(A) = val_{D,\gamma}(A).$$
  3. Ist $A \in For_\Sigma$ geschlossen, dann gilt
  $$val_{D,\beta}(A) = val_{D,\gamma}(A)$$
  für alle Belegungen β, γ, d.h. der Wert von A hängt nicht von der Belegung ab.

  $\looparrowright$ [Lemma 4.40, Skript S. 100]

- Σ sei eine Signatur, D eine Interpretation für Σ, β eine Belegung, σ eine Substitution und $t \in Term_\Sigma$. Dann gilt
  $$val_{D,\beta}(\sigma(t)) = val_{D,\beta}(t)$$
  wobei $\beta'(x) = val_{D,\beta}(\sigma(x))$ für alle $x \in Var$.
  $\looparrowright$ [Lemma 4.41, Skript S. 101]

- Σ sei eine Signatur, D eine Interpretation für Σ, β eine Belegung, $A \in For_\Sigma$ und σ eine für A kollisionsfreie Substitution. Dann gilt
  $$val_{D,\beta}(\sigma(A)) = val_{D,\beta}(A)$$
  wobei $\beta'(x) = val_{D,\beta}(\sigma(x))$ für alle $x \in Var$.
  $\looparrowright$ [Lemma 4.42, Skript S. 101]

- Sei Σ eine Signatur, D eine Interpretation für Σ, β eine Belegung und σ eine für A kollisionsfreie Substitution mit $\sigma(y) = y$ für alle Variablen $y \neq x$, dann gilt

  - $val_{D,\beta}(\forall x A \to \sigma(A)) = W$
  - $val_{D,\beta}(\sigma(A) \to \exists x A) = W$.

  $\looparrowright$ [Lemma 4.44, Skript S. 103] Eine Interpretation D über Σ heisst **Modell** einer Formel A über Σ, wenn für jedes β gilt $val_{D,\beta}(A) = W$. D heisst Modell einer Formelmenge M, wenn für jedes β und jede Formel $B \in M$ gilt $val_{D,\beta}(B) = W$.
  $\looparrowright$ [Definition 4.45, Skript S. 103]

- Es sei $M \subseteq For_\Sigma$, $A \in For_\Sigma$.
  $$M \models_\Sigma A \quad :\Leftrightarrow \quad \text{Jedes Modell von } M \text{ ist auch Modell von } A.$$
  Lies: **Aus** M **folgt** A (über Σ). Wir werden meistens kurz $\models$ statt $\models_\Sigma$ schreiben. Ferner stehe wieder $\models A$ für $\emptyset \models A$ und $B \models A$ für $\{B\} \models A$.
  $\looparrowright$ [Definition 4.46, Skript S. 104]

- Anders als in der Aussagenlogik gilt $A \models B \Leftrightarrow \models A \to B$ (Korrolar 2.17) im allgemeinen nicht mehr. $\models A \to B \Rightarrow A \models B$ ist zwar weiterhin gültig, nicht aber die Umkehrung.
  $\looparrowright$ [Bemerkung 4.47, Skript S. 104]

•

$$
\begin{array}{rll}
D \text{ ist Modell von } A & \text{gdw} & D \text{ ist Modell von } Cl_\forall A \\
D \text{ ist Modell von } \neg A & \text{gdw} & D \text{ ist Modell von } \neg Cl_\exists \neg A \\
M \models A & \text{gdw} & Cl_\forall M \models A \\
& \text{gdw} & M \models Cl_\forall A \\
& \text{gdw} & Cl_\forall M \models Cl_\forall A \\
& \text{gdw} & M \cup \{\neg Cl_\forall A\} \text{ hat kein Modell}
\end{array}
$$

↬ [Lemma 4.48, Skript S. 104]

- $A \in For_\Sigma$ heisst
  - **allgemeingültig** gdw $\models A$
  - **erfüllbar** gdw $\neg A$ nicht allgemeingültig ist.

↬ [Definition 4.49, Skript S. 104]

- 1. Die folgenden Aussagen sind äquivalent:
     (a) A ist allgemeingültig
     (b) Jede Interpretation D ist Modell von A
     (c) $val_{D,\beta}(A) = W$ für alle Interpretationen D und Variablenbelegungen β
     (d) $Cl_\forall A$ ist allgemeingültig
  2. Die folgenden Aussagen sind äquivalent:
     (a) A ist erfüllbar
     (b) Es gibt eine Interpretation D und eine Belegung β mit $val_{D,\beta}(A) = W$
     (c) $Cl_\exists A$ hat ein Modell
  3. Falls M keine freien Variablen enthält gilt

$$M \cup \{\neg A\} \text{ ist nicht erfüllbar} \quad \text{gdw} \quad M \models A.$$

↬ [Lemma 4.50, Skript S. 104]

- $A \in For_\Sigma$ heisst **Tautologie**, wenn es eine endliche aussagenlogische Signatur $\Sigma' = \{P_0, \ldots, P_{n-1}\}$, ein $A' \in For_{\Sigma'}$ und Formeln $A_0, \ldots, A_{n-1} \in For_\Sigma$ gibt, so dass
  - A' ist (aussagenlogisch) allgemeingültig über Σ'
  - A entsteht aus A', indem man dort P_i durch A_i ersetzt (für $i = 0, \ldots, n-1$).

↬ [Definition 4.51, Skript S. 105]

- Jede Tautologie ist allgemeingültig
  ↬ [Lemma 4.53, Skript S. 105]

- A, B heissen **logisch äquivalent** gdw $\models A \leftrightarrow B$ (d.h. $A \leftrightarrow B$ ist allgemeingültig).
  ↬ [Definition 4.54, Skript S. 105]

- Logische Äquivalenz ist eine Äquivalenzrelation auf For_Σ. Sie ist darüberhinaus eine Kongruenz, d.h.

$$
\begin{array}{rl}
\text{gilt} & \models A \leftrightarrow A', \ \models B \leftrightarrow B' \\
\text{dann auch} & \\
& \models \neg A \leftrightarrow \neg A' \\
& \models (A\, op\, B) \leftrightarrow (A'\, op\, B') \text{ für } op \in \{\wedge, \vee, \rightarrow, \leftrightarrow\} \\
& \models QxA \leftrightarrow QxA' \text{ für } x \in Var,\ Q \in \{\forall, \exists\}
\end{array}
$$

↬ [Korollar 4.55, Skript S. 106]

- Es seien $A, A', B, B' \in For_\Sigma$ mit
  - B ist Teilformel von A,
  - B ist logisch äquivalent zu B',
  - A' entsteht aus A, indem man dort an irgendwelchen Stellen (nicht notwendig überall) B durch B' ersetzt.

  Dann ist A logisch äquivalent zu A'.
  ↬ [Korollar 4.56, Skript S. 106]

2.2 Normalformen

- Eine Formel $A \in For_\Sigma$ ist in **Negationsnormalform**, wenn jedes Negationszeichen in A vor einer atomaren Teilformel steht.
  ↬ [Definition 4.71, Skript S. 111]
- Eine Formel $A \in For_\Sigma$ heisst **bereinigt**, wenn
  - $Frei(A) \cap Bd(A) = \emptyset$
  - die hinter Quantoren stehenden Variablen paarweise verschieden sind.

  ↬ [Definition 4.73, Skript S. 111]
- $A \in For_\Sigma$ hat **Pränex-Normalform**, wenn A die Gestalt hat

  $$Q_1 x_1 \dots Q_n x_n B \quad \text{mit} \quad Q_i \in \{\forall, \exists\},\, x_i \in Var\,(i = 1, \dots, n)$$

  und: B ist eine quantorenfreie Formel. Man nennt B auch die **Matrix** von A.
  ↬ [Definition 4.75, Skript S. 111]
- Gegeben sei eine geschlossene Formel über Σ der Gestalt

  $$\forall x_1 \dots \forall x_n \exists y B,$$

  in der $x_1, \dots, x_n, y$ paarweise verschieden sind und $x_1, \dots, x_n \notin Bd(B)$ ($n = 0$ ist erlaubt, die Formel ist dann $\exists y B$). Wir erweitern Σ um ein neues, n-stelliges Funktionssymbol g zu einer Signatur Σ_g (d.h. $F_{\Sigma_g} = F_\Sigma \cup \{g\}$). Dann gilt

  $$\forall x_1 \dots \forall x_n \exists y B \text{ hat ein Modell über } \Sigma \quad \Leftrightarrow \quad \begin{array}{l} \forall x_1 \dots \forall x_n \{y/g(x_1, \dots, x_n)\}(B) \\ \text{hat ein Modell über } \Sigma_g \end{array}$$

  ↬ [Lemma 4.81, Skript S. 113]
- Eine Formel ist in **Skolem-Normalformel**, wenn sie
  - geschlossen ist,
  - die Gestalt $\forall x_1 \dots \forall x_n B$ (nur Allquantoren im Präfix), mit quantorenfreiem B,
  - und die Matrix B in KNF ist.

  ↬ [Definition 4.82, Skript S. 114]
- Die bei Anwendung von Lemma 4.81 neu eingeführten Funktionssymbole nennt man **Skolemfunktionen**, im nullstelligen Fall **Skolemkonstanten**. Das Verfahren selber heisst **Skoleminisierung**. Man beachte, dass die neue Signatur Σ_{sk} von der vorgelegten Formel A abhängt.
  ↬ [Bemerkung, Skript S. 115]
- Die Signatur Σ enthalte mindestens eine Konstante. Eine Interpretation (D, I) von Σ heisst **Herbrand-Interpretation** oder **Herbrand-Algebra** genau dann, wenn

1. $D = Term^0_\Sigma$[1]
2. $I(f)(t_1, \dots, t_n) = f(t_1, \dots, t_n)$ für alle n-stelligen Funktionssymbole f, und beliebige Grundterme $t_1, \dots, t_n$. (Im Fall $n = 0$ heisst das: $I(c) = c$)

Zu einer Formelmenge $M \subseteq For_\Sigma$ ist ein **Herbrand-Modell** von M eine Herbrand-Algebra über Σ, welche Modell von M ist.
In einer Herbrand-Algebra wird also jeder Grundterm als er selbst interpretiert,

$$I(t) = t \text{ für Grundterme,}$$

insbesondere gilt das für Konstanten. Spielraum für verschiedene Herbrand-Algebren gibt es nur bei der Interpretation der Prädikatsymbole.
$\looparrowright$ [Definition 4.86, Skript S. 116]

- Ist $(Term^0_\Sigma, I)$ Herbrand-Algebra über Σ, $t \in Term_\Sigma$ mit $Var(t) = \{x_1, \dots, x_n\}$, und β eine Belegung der Variablen nmit Grundtermen, dann ist

$$val_{D,I,\beta}(t) = \{x_1/\beta(x_1), \dots, x_n/\beta(x_n)\}(t).$$

$\looparrowright$ [Korollar 4.87, Skript S. 116]

- Sei $A := \forall x_1 \dots \forall x_n B$ mit quantoremfreiem B eine geschlossene Formel. Eine **Grundinstanz** von A ist eine Formel

$$\{x_1/t_1, \dots, x_n/t_n\}(B) \quad \text{mit} \quad t_1, \dots, t_n \in Term^0_\Sigma.$$

Ist M eine Menge geschlossener, universell quantifizierter Formeln, so sei

$$\text{Grundinstanz}(M)$$

die Menge aller Grundinstanzen von Formeln in M.
$\looparrowright$ [Definition 4.88, Skript S. 116]

- Σ enthalte mindestens eine Konstante, und es sei M eine Menge geschlossener, universell quantifizierter Formeln. Ferner enthalte keine Formel in M das Gleichheitssymbol $\doteq$. Dann sind äquivalente Aussagen:

1. M hat ein Modell
2. M hat ein Herbrand-Modell
3. Grundinstanzen(M) hat ein Modell
4. Grundinstanzen(M) hat ein Herbrand-Modell.

$\looparrowright$ [Satz 4.89, Skript S. 117]

- Ein **Literal** ist ein Atom oder ein negiertes Atom. Es heisst positives Literal im ersten und negatives im zweiten Fall. Ein **Grundliteral** ist ein Literal ohne Variable. Ein **komplementäres Paar** von Literalen ist ein Paar $\{L, \neg L\}$ (L Atom).
$\looparrowright$ [Definition 4.91, Skript S. 118]

- Für eine Konjunktion $L_1 \wedge \dots \wedge L_k$ $k \geq 1$, von Grundliteralen gilt:

1. $L_1 \wedge \dots \wedge L_k$ ist nicht allgemeingültig
2. $L_1 \wedge \dots \wedge L_k$ hat ein Modell $\Leftrightarrow$ $\{L_1, \dots, L_k\}$ enthält kein komplementäres Paar

Für eine Disjunktion $L_1 \vee \dots \vee L_k$, $k \geq 1$ von Grundliteralen gilt:

1. $L_1 \vee \dots \vee L_k$ hat ein Modell
2. $L_1 \vee \dots \vee L_k$ ist allgemeingültig $\Leftrightarrow$ $\{L_1, \dots, L_k\}$ enthält ein komplementäres Paar

$\looparrowright$ [Lemma 4.92, Skript S. 118]

[1] $Term^0_\Sigma$ sei die Menge aller Grundterme über Σ

2.3 Beweistheorie

- Prädikatenlogische Vorzeichenformeln sind prädikatenlogische Formeln, vor denen 0 oder 1 als Vorzeichen stehen. Vorzeichenformeln werden wieder in Typen eingeteilt, und entsprechend dieser Einteilung werden Abkömmlinge definiert:
  Uniforme Notation

  Typ ϵ: $1A$, $0A$ für Atome A

  Typ α:

V	V_1	V_2
$1\neg A$	$0A$	–
$0\neg A$	$1A$	–
$1A \wedge B$	$1A$	$1B$
$0A \vee B$	$0A$	$0B$
$0A \rightarrow B$	$1A$	$0B$

  Typ β:

V	V_1	V_2
$0A \wedge B$	$0A$	$0B$
$1A \vee B$	$1A$	$1B$
$1A \rightarrow B$	$0A$	$1B$

  Typ γ:

V	V_1
$1\forall x A(x)$	$1A(x)$
$0\exists x A(x)$	$0A(x)$

  Typ δ:

V	V_1
$1\exists x A(x)$	$1A(x)$
$0\forall x A(x)$	$0A(x)$

  $\looparrowright$ [Skript S. 121] Sei (D, I) eine beliebige Interpretation, β eine beliebige Variablenbelegung für (D, I). Wir schreiben val_β statt $val_{D,I,\beta}$. Dann gilt

  1. für jede α-Formel V:
  $$val_\beta(V) = W \quad \Leftrightarrow \quad val_\beta(V_1) = W \text{ und } val_\beta(V_2) = W$$
  2. für jede β-Formel V:
  $$val_\beta(V) = W \quad \Leftrightarrow \quad val_\beta(V_1) = W \text{ und } val_\beta(V_2) = W$$
  3. für jede γ-Formel V:
  $$val_\beta(V) = W \quad \Leftrightarrow \quad \text{Für alle } d \in D \;:\; val_{\beta_x^d}(V_1(x)) = W$$
  4. für jede δ-Formel V:
  $$val_\beta(V) = W \quad \Leftrightarrow \quad \text{Es gibt } d \in D \text{ mit: } val_{\beta_x^d}(V_1(x)) = W$$

  $\looparrowright$ [Lemma 5.1, Skript S. 121]

- Sei A eine Formel und M eine Menge von Formeln, so wird ein **Tableau für** A **über** M entsprechend dem aussagenlogischen Fall (Def. 3.31) definiert.

  1. Ein einziger mit $0Cl_\forall A$ markierter Knoten ist ein Tableau für A über M.
  2. α-Regel (wie im AL-Fall).
  3. β-Regel (wie im AL-Fall).
  4. Es sei T ein Tableau für A über M und π ein Pfad in T, so dass auf π ein Knoten liegt, der mit einem V vom Typ γ beschriftet ist, $V = 1\forall x B_1(x)$ oder $V = 0\exists x B_1(x)$. Dann ist auch T_1 ein Tableau für A über M, welches dadurch aus T entsteht, dass an π ein neuer Knoten angefügt wird, welcher mit $V_1(y)$ (d.h. $1B_1(y)$ bzw. $0B_1(y)$) beschriftet ist für eine Variable y, die noch nicht auf π frei vorkommt. Wir sagen in diesem Fall, dass T_1 aus T durch die Anwendung der **γ-Regel** hervorgeht.
  5. Es sei T ein Tableau für A über M und π ein Pfad in T, so dass auf π ein Knoten liegt, der mit einem V vom Typ δ beschriftet ist, $V = 0\forall x B_1(x)$ oder $V = 1\exists x B_1(x)$. Dann ist auch T_2 ein Tableau für A über M, welches dadurch aus T entsteht, dass an π ein Knoten angefügt wird, welcher mit $V_1(f(x_1, \ldots, x_n))$ beschriftet ist, wobei $x_1, \ldots, x_n$ alle freien Variablen in V sind und f ein neues Funktionszeichen ist. Wir sagen T_2 geht aus T durch eine Anwendung der **δ-Regel** hervor.

6. Es sei T ein Tableau für A über M und π ein Pfad in T, auf welchem die Formeln $1t \doteq s$ und V vorkommen, wobei t' ein in V vorkommender Teilterm ist. Ausserdem existiere eine Substitution σ mit $\sigma(t') = \sigma(t)$. Dann ist auch T_4 ein Tableau für A über M, welches durch Verlängerung von π um einen mit V' beschrifteten Knoten und Anwendung von σ auf das ganze Tableau entsteht, wobei V' aus $\sigma(V)$ entsteht, indem ein Vorkommen von $\sigma(t)$ durch $\sigma(s)$ ersetzt wird. Wir sagen, T_4 geht aus T durch eine Anwendung der G_l**-Regel** aus T hervor.
7. Analog für die G_r**-Regel**.
8. Es sei T ein Tableau für A über M und π ein Pfad in T, dann ist auch T_3 ein Tableau für A über M, welches durch Verlängerung von π um einen mit $1Cl_\forall B$ beschrifteten Knoten entsteht, für eine Formel B aus M. Wir sagen, T_3 geht aus T durch eine Anwendung der V**-Regel** aus T hervor.

Zusammenfassung der Tableauregeln in Kurzform:

α-Regel:	$\dfrac{V}{\begin{matrix}V_1\\V_2\end{matrix}}$	für α-Formeln V
β-Regel:	$\dfrac{V}{V_1 \mid V_2}$	für β-Formeln V
γ-Regel:	$\dfrac{V}{V_1(y)}$	für γ-Formeln V und jede Variable y, die auf dem Pfad noch nicht vorkommt
δ-Regel:	$\dfrac{V}{V_1(f(x_1,\ldots,x_n))}$	für δ-Formeln V, wobei $x_1,\ldots,x_n$ alle freien Variablen in V sind und f ein neues n-stelliges Funktionssymbol ist
G_l-Regel:	$\dfrac{1t \doteq s \quad V(t')}{\sigma V(s)}$	wobei $\sigma(t') = \sigma(t)$
G_r-Regel:	$\dfrac{1t \doteq s \quad V(s')}{\sigma V(t)}$	wobei $\sigma(s') = \sigma(s)$
Anfangsregel:	$\dfrac{}{0Cl_\forall A}$	für die zu beweisende Formel A
V-Regel:	$\dfrac{}{1Cl_\forall B}$	für jedes $B \in M$

Die Substitution σ in der G_l- und G_r-Regel wird jeweils auf das ganze Tableau angewendet.
↬ [Definition 5.2, Skript S. 122]

- Es sei π ein Pfad in T und σ eine Substitution. σ **schliesst** π, wenn es
  - Formeln B, C gibt, so dass $\sigma(B) = \sigma(C)$, σ kollisionsfrei für B und C ist und $1B$, $0C$ auf π liegen oder
  - Terme s, t gibt, so dass $\sigma(s) = \sigma(t)$ und $0s \doteq t$ auf π liegt.

  Die **Schliessregel** oder **C-Regel** ist:
  Aus einem Tableau T erzeuge ein Tableau T_1 dadurch, dass für einen Pfad π und eine Substitution σ, die π schliesst, σ auf ganz T angewandt wird.
  Nach Anwendung der C-Regel ist der zugehörige Pfad π geschlossen. Ein Tableau T heisst geschlossen, wenn alle seine Pfade geschlossen sind.
  ↬ [Definition 5.3, Skript S. 123]

- Der **prädikatenlogische Tableaukalkül T** ist definiert wie folgt. **T** hat keine Axiome (**T** wird nämlich als Widerlegungskalkül eingesetzt). **T** arbeitet auf Tableaux als Objekten für die Regeln. Die Regeln von **T** stellen jeweils aus einem Tableau T ein Tableau T_1 her gemäß einer der Regeln (α, β, γ, δ, G_l, G_r und C) anzuwenden auf die Vorzeichenformeln V bzw. Pfade π in T.
  Gegeben sei eine Formelmenge M und eine Formel A. A ist ableitbar aus M in **T**,

$$M \vdash_T A \quad \text{gdw} \quad \text{ein geschlossenes Tableau } T \text{ für } A \text{ über } M \text{ existiert.}$$

↬ [Definition 5.5, Skript S. 123]

- Es seien $A \in For_\Sigma$, $M \subseteq For_\Sigma$, T ein Tableau für A über M und (D, I) eine Interpretation über $\overline{\Sigma}$, wobei $\overline{\Sigma} = \Sigma \cup \{f \mid f$ neues Funktionssymbol in $T\}$. (D, I) heisst **Modell von T über M** gdw. gilt
  - (D, I) ist Modell von M
  - zu jeder Variablenbelegung β gibt es einen Pfad π in T mit $val_{D,I,\beta}(V) = W$ für alle V auf π.

  ↬ [Definition 5.7, Skript S. 127]

- Ist D Modell von T über M und entsteht T' aus T durch Schliessen eines Pfades, dann ist D auch Modell von T'.
  ↬ [Lemma 5.9, Skript S. 129]

- Sei $A \in For_\Sigma$, $M \subseteq For_\Sigma$. Wenn es ein geschlossenes Tableau für $Cl_\forall A$ über $Cl_\forall M$ gibt, dann ist auch $M \models A$.
  ↬ [Satz 5.10, Skript S. 129]

- Sei A eine Formel und M eine Menge von Formeln. Gilt $M \models A$, dann gibt es ein geschlossenes Tableau für $Cl_\forall A$ über $Cl_\forall M$.
  ↬ [Satz 5.15, Skript S. 132]

- Der **Hilbertkalkül** (für eine Signatur Σ) bezeichnet mit **H** wird durch die folgenden Axiome und Regeln gegeben:
  (Wie in der Aussagenlogik sind die Regeln – insbesondere die Axiome – als Schemata angegeben. $x, y, z, \ldots$ stehen für Variablen, f für ein Funktionssymbol, p für ein Relationssymbol, t für einen Term, A, B, C stehen für Formeln.)
  Axiome

Ax1	$A \to (B \to A)$	(Abschwächung)
Ax2	$(A \to (B \to C)) \to ((A \to B) \to (A \to C))$	(Verteilung von $\to$)
Ax3	$(\neg A \to \neg B) \to (B \to A)$	(Kontraposition)
Ax4	$\forall x A \to \{x/t\}(A)$ wobei $\{x/t\}$ kollisionsfrei für A	(Instantiierung)
Ax5	$\forall x(A \to B) \to (A \to \forall x B)$ wobei $x \notin Frei(A)$	($\forall$-Verschiebung)
Gl1	$x \doteq x$	(Reflexivität)
Gl2	$x \doteq y \to y \doteq x$	(Symmetrie)
Gl3	$x \doteq y \to (y \doteq z \to x \doteq z)$	(Transitivität)
Gl4	$x_1 \doteq y_1 \to (x_2 \doteq y_2 \to (\ldots(x_n \doteq y_n \to f(x_1, \ldots, x_n) \doteq f(y_1, \ldots, y_n))\ldots))$	
Gl5	$x_1 \doteq y_1 \to (x_2 \doteq y_2 \to (\ldots(x_n \doteq y_n \to p(x_1, \ldots, x_n) \doteq p(y_1, \ldots, y_n))\ldots))$	

  Regeln

  $$\text{Mp:} \quad \frac{A, A \to B}{B} \quad \text{(Modus ponens)}$$
  $$\text{Gen:} \quad \frac{A}{\forall x A} \quad \text{(Generalisierung)}$$

  Ableitbarkeit im Hilbertkalkül wird bezeichnet mit $\vdash_H$.
  ↬ [Definition 5.27, Skript S. 138]

- Ein **Literal** ist eine atomare oder negierte atomare Formel. Eine **Klausel** ist eine endliche Menge von Literalen. Die **leere Klausel** wird mit $\square$ bezeichnet.
  ↬ [Definition 5.30, Skript S. 139]

- Ist A eine quantorenfreie Formel und σ eine Variablenumbenennung, dann heisst $\sigma(A)$ eine **Variante** von A.
  ↬ [Definition 5.31, Skript S. 139]

- Über der Menge der Klauseln (über der prädikatenlogischen Signatur Σ) ist die **Resolution** die folgende Regel:
  $$\frac{C_1 \cup K_1 \qquad C_2 \cup K_2}{\mu(C_1 \cup C_2)}$$
  wobei gilt:

– C_1, C_2, K_1, K_2 sind Klauseln

– $K_1, K_2 \neq \Box$

– $Var(C_1 \cup K_1) \cap Var(C_2 \cup K_2) = \emptyset$

– μ ist allgemeinster Unifikator von $K_1 \cup \sim K_2$.

Eine Klausel C heisst Resolvente zweier Klauseln, wenn sie durch Anwendung von Res auf diese entsteht.
↬ [Definition 5.32, Skript S. 140]

- Sei M eine Klauselmenge.

  1. Mit $Res(M)$ bezeichnen wir die Menge aller Resolventen von Klauseln aus M. Genauer:

  $$Res(M) = \left\{ B \;\middle|\; \begin{array}{l} \text{es gibt Varianten } C_1, C_2 \text{ von Klauseln aus } M, \\ \text{so dass } B \text{ eine Resolvente von } C_1, C_2 \text{ ist.} \end{array} \right\}$$

  2. $R^n(M) = \underbrace{Res(\ldots(Res(}_{n-mal}M)\ldots)$

  3. $M \vdash_R A$ gilt genau dann, wenn es ein n gibt mit $A \in Res^n(M)$

↬ [Definition 5.33, Skript S. 140]

- **Verfahren:**
Gegeben: eine prädikatenlogische Formel F (mit eventuellen Vorkommen von freien Variablen).

  1. Bereinige F durch systematisches Umbenennen der gebundenen Variablen. Es entsteht eine zu F äquivalente Formel F_1.
  2. Seien $y_1, y_2, \ldots, y_n$ die in F bzw. F_1 vorkommenden freien Variablen. Ersetze F_1 durch $F_2 = \exists y_1 \exists y_2 \ldots \exists y_n F_1$. Dann ist F_2 erfüllbarkeitsäquivalent zu F_1 und F und enthält keine freien Variablen mehr.
  3. Stelle eine zu F_2 äquivalente (und damit zu F erfüllbarkeitsäquivalente) Aussage F_3 in Pränexform her.
  4. Eliminiere die vorkommenden Existenzquantoren durch Übergang zur Skolemform von F_3. Diese sei F_4 und ist dann erfüllbarkeitsäquivalent zu F_3 und damit auch zu F.
  5. Forme die Matrix von F_4 um in KNF (und schreibe die Formel F_5 dann als Klauselmenge auf).

↬ [Schöning S. 66]
↬ [Skript S. 141]

- Eine **Sequenz** wird notiert als eine Folge zweier endlicher Mengen prädikatenlogischer Formeln getrennt durch das Symbol $\Rightarrow$:

$$\Gamma \rightarrow \Delta.$$

Γ wird Antezedent und Δ Sukzedent genannt. Sowohl links wie rechts vom Sequenzenpfeil $\Rightarrow$ kann auch die leere Folge stehen.
↬ [Definition 5.41, Skript S. 145]

- Sei D eine prädikatenlogische Struktur und β eine Variablenbelegung:

$$val_{D,\beta}(\Gamma \Rightarrow \Delta) = val_{D,\beta}(\bigwedge \Gamma \rightarrow \bigvee \Delta)$$

Es gelten die üblichen Vereinbarungen für leere Disjunktionen und Konjunktionen.
↬ [Definition 5.42, Skript S. 145]

- Die Regeln des Sequenzenkalküls sind in den Abbildungen 5.1, 5.2 und 5.3 (Skript S. 146/147) zusammengestellt.

2.4 Weitere Anmerkungen zur Prädikatenlogik erster Ordnung

- **Kompaktheit der PL1**: Für beliebige $M \subseteq For_\Sigma$, $A \in For_\Sigma$ gilt:

$$M \models A \quad \Leftrightarrow \quad E \models A \text{ für eine endliche Teilmenge } E \text{ von } M.$$

$\looparrowright$ [Satz 5.49, Skript S. 150]

- **Endlichkeitssatz**: Eine Menge $M \subseteq For_\Sigma$ hat genau dann ein Modell, wenn jede endliche Teilmenge von M ein Modell hat. (Spezialfall von Satz 5.49)
$\looparrowright$ [Satz 5.50, Skript S. 150]

- Es sei Σ eine Signatur der PL1. Dann gibt es keine Formelmenge $M \subseteq For_\Sigma$, so dass für jede Interpretation (D, I) über Σ gilt:

$$(D, I) \textbf{ ist Modell von } M \quad \Leftrightarrow \quad D \text{ ist endlich.}$$

$\looparrowright$ [Satz 5.51, Skript S. 150]

- Eine **ordnungssortierte Signatur** Σ ist ein Tupel

$$\Sigma = (S_\Sigma, \leq, F_\Sigma, P_\Sigma, \alpha_\Sigma),$$

wo $S_\Sigma, F_\Sigma, P_\Sigma$ paarweise disjunkte Mengen von Symbolen sind und ferner gilt:

1. S_Σ besteht aus endlich vielen Sortensymbolen;
2. $\leq$ ist eine partielle Ordnung auf der Menge S_Σ;
3. F_Σ, P_Σ sind wie früher die Mengen der Funktions- bzw. Prädikatsymbole;
4. α_Σ ist jetzt nicht mehr einfach die Stelligkeitsfunktion, sondern gibt zu jedem Funktions- bzw. Prädikatsymbol seine Sortierung an:

$$\alpha_\Sigma : F_\Sigma \cup P_\Sigma \to S_\Sigma^* \text{ wobei } \alpha_\Sigma(f) \in S_\Sigma^+, \text{ wenn } f \in F_\Sigma.$$

$\alpha_\Sigma(f) = Z_1 \dots Z_n Z'$ bedeutet: f ist ein Funktionssymbol für Funktionen, welche n-Tupeln von Elementen der Sorten $Z_1, \dots, Z_n$ (respective) ein Element der Sorte Z' zuordnen.

$\alpha_\Sigma(p) = Z_1, \dots, Z_n$ bedeutet: p ist gedacht zur Notation von Mengen von n-Tupeln von Elementen respective der Sorten $Z_1, \dots, Z_n$. I Fall $n = 0$ ist (wie früher) p aussagenlogisches Atom.

Man nimmt ferner an, dass auch die Mengen der Variablen sortiert ist: Zu jedem $Z \in S_\Sigma$ gibt es unendlich viele Variablen der Sorte Z, und für verschiedene Sorten sind diese Variablenmengen disjunkt.

$\looparrowright$ [Definition 5.53, Skript S. 152]

- Eine **Interpretation** über der sortierten Signatur $\Sigma = (S_\Sigma, F_\Sigma, P_\Sigma, \alpha_\Sigma)$ ist ein Paar (D, I), wo gilt

$$D = \{D_Z \mid Z \in S_\Sigma\} \text{ ist eine Familie nichtleerer (den Sorten } Z \in S_\Sigma \text{ entsprechender) Mengen,}$$

I ordnet jedem Funktions- und Prädikatsymbol eine sortengerechte Bedeutung zu:

$$I(f) : D_{Z_1} \times \dots \times D_{Z_n} \to D_{Z'} \text{ wenn } \alpha(f) = Z_1 \dots Z_n Z'$$
$$I(p) \subseteq D_{Z_1} \times \dots \times D_{Z_n} \text{ wenn } \alpha(p) = Z_1 \dots Z_n.$$

$\looparrowright$ [Definition 5.54, Skript S. 153]

- Die Sortierung wird auf **Terme** übertragen:
  - Ist x eine Variable der Sorte Z, dann ist x Term von der Sorte Z
  - Sind $t_1, \dots, t_n$ Terme der Sorten respective $Z_1, \dots, Z_n$ und f ein Funktionssymbol mit $\alpha_f = Z_1 \dots Z_n Z'$, dann ist $f(t_1, \dots, t_n)$ ein Term der Sorte Z'.

Entsprechend geschieht die sortengerechte Definition der Formeln.
$\looparrowright$ [Definition 5.55, Skript S. 153]

2.5 Anwendung

- Eine Algebra mit einer einstelligen Funktion $\neg$ und einer zweistelligen Funktion $\vee$ heisst eine **Robbins Algebra**, wenn sie die folgenden Axiome erfüllt:

$$\begin{array}{ll} \text{R1} & x \vee y = y \vee x \\ \text{R2} & (x \vee y) \vee z = x \vee (y \vee z) \\ \text{R3} & \neg[\neg(x \vee y) \vee \neg(x \vee \neg y)] = x \end{array}$$

  $\looparrowright$ [Definition 5.56, Skript S. 155]

- In jeder Robbins Algebra gilt

$$\forall x \exists z (\neg z = x)$$

  $\looparrowright$ [Lemma 5.57, Skript S. 156]

- Ist in einer Robbins Algebra die Formel

$$\forall x, y \; \neg x = \neg y \to x = y$$

  wahr, dann ist sie eine Boolesche Algebra.
  $\looparrowright$ [Lemma 5.58, Skript S. 156]

- Jede endliche Robbins Algebra ist eine Boolesche Algebra.
  $\looparrowright$ [Satz 5.59, Skript S. 156]

- Falls es in einer Robbins Algebra R ein Element a gibt, so dass

$$\forall x \, (a \vee x = x)$$

  gilt, dann ist R eine Boolesche Algebra.
  $\looparrowright$ [Lemma 5.60, Skript S. 157]

- Wenn eine der beiden folgenden Bedingungen in einer Robbins ALgebra erfüllt sind, dann ist sie schon eine Boolesche Algebra.

  1. $\exists x \, (\neg(x \vee x) = \neg x)$
  2. $\exists x, y \, (\neg(x \vee y) = \neg y)$

  $\looparrowright$ [Lemma 5.61, Skript S. 158]

3 Die Behandlung der Gleichheit

3.1 Abstrakte Datentypen

- 1. Zu einer Menge E von Gleichungen über einer Signatur Σ definieren wir für Terme t, s:

$$s \xrightarrow{1}_E t$$

     genau dann, wenn es eine Gleichung $l \doteq r \in E$ gibt und eine Substitution σ, so dass gilt
     - $\sigma(l)$ ist Unterterm von s
     - t entsteht aus s, indem dort der Unterterm $\sigma(l)$ an genau einer Stelle ersetzt wird durch $\sigma(r)$.

     Man beachte, dass nur die linke Seite durch die rechte ersetzt wird.
  2. $\to_E$ ist die reflexive, transitive Hülle von $\xrightarrow{1}_E$
  3. $\leftrightarrow_E$ ist die reflexive, transitive, symmetrische Hülle von $\xrightarrow{1}_E$

  Wenn das zugehörige Gleichungssystem klar ist, schreiben wir einfach $\xrightarrow{1}$, $\to$, $\leftrightarrow$.
  $\looparrowright$ [Definition 6.1, Skript S. 161]

- 1. für $l \doteq r$ in E gilt trivialerweise $l \to_E r$.
  2. $s \to_E t \Rightarrow s \leftrightarrow_E t$
  3. $s \xrightarrow{1}_E t \Rightarrow \sigma(s) \xrightarrow{1}_E \sigma(t)$ für jede Substitution σ.
  4. $s \leftrightarrow_E t \Rightarrow \sigma(s) \leftrightarrow_E \sigma(t)$ für jede Substitution σ.
  5. $s \leftrightarrow_E t \Rightarrow E \models s \doteq t$

  ↬ [Lemma 6.2, Skript S. 161]

- **Freie Algebra**: Sei E eine Menge von Gleichungen über der Signatur Σ. Die freie Algebra zu Σ, E ist die Interpretation
$$F_E = (\overline{T}_{\Sigma,E}, Frei_E)$$
mit
$$\overline{T}_{\Sigma,E} = \{[t]_E \mid t \in Term_\Sigma\} \quad \text{und} \quad Frei_E(f)([t_1]_E, \dots, [t_n]_E) = [f(t_1, \dots, t_n)]_E.$$
Die freie Algebra wird gelegentlich auch freie Interpretation oder freies Modell von E genannt.
↬ [Definition 6.3, Skript S. 161]

- **Initiale Algebra**: Sei E eine Menge von Gleichungen über der Signatur Σ, die mindestens ein Konstantensymbol enthält.
Die initiale Algebra zu Σ, E ist die Interpretation
$$I_{\Sigma,E} = (\overline{T}^0_E, Init_E)$$
mit
$$\text{Universum } \overline{T}^0_{\Sigma,E} = \{[t]^0_E \mid t \in Term^0_\Sigma\}$$
und
$$\text{Interpretation } Init_E(f)([t_1]^0_E, \dots, [t_n]^0_E) = [f(t_1, \dots, t_n)]^0_E.$$
Die initiale Algebra wird gelegentlich auch initiale Interpretation oder initiales Modell von E genannt.
Die Voraussetzung, dass Σ mindestens eine Konstante enthalte, ist notwendig um sicherzustellen, dass das Universum von $I_{\Sigma,E}$ nicht leer ist.
↬ [Definition 6.4, Skript S. 162]

- Es sei t ein Term mit $Var(t) \subseteq \{x_1, \dots, x_n\}$. Der Einfachheit halber schreiben wir T für $T_{\Sigma,E}$, T^0 für $T^0_{\Sigma,E}$, und I für $Frei_E$ und $Init_E$.
  1. Für jede Belegung β mit $\beta(x_i) = [s_i]_E$, $i = 1, \dots, n$ gilt:
  $$val_{T,\beta}(t) = [\{x_1/s_1, \dots, x_n/s_n\}(t)]_E$$
  2. Entsprechend für T^0
  3. Ist t ein Grundterm, dann gilt
  $$I(t) = [t]^0_E.$$

  ↬ [Lemma 6.5, Skript S. 162]

- $T_{\Sigma,E}$ und $T^0_{\Sigma,E}$ sind Modelle von E.
↬ [Lemma 6.6, Skript S. 162]

- Sei E eine Menge von Gleichungen in der Signatur Σ.
  1. Für eine Gleichung $s \doteq t$ sind äquivalente Aussagen:
     (a) $s \leftrightarrow_E t$
     (b) $E \models s \doteq t$
     (c) $s \doteq t$ gilt im freien Modell von E.
  2. Enthält Σ mindestens eine Konstante und ist $s \doteq t$ eine Gleichung zwischen Grundtermen, so sind diese Aussagen zusätzlich äquivalent zu:

(d) $s \doteq t$ gilt im initialen Modell von E.

↬ [Satz 6.7, Skript S. 163]

- Sei E eine Menge von Gleichungen in der Signatur Σ.

  1. Es sei D ein beliebiges Modell von E und β eine Belegung der Variablen mit Elementen von D. Dann gibt es genau einen Homomorphismus ϕ vom freien Modell von E nach D, so dass für alle Variablen x gilt
  $$\phi([x]_E) = \beta(x)$$
  2. Σ enthalte mindestens eine Konstante. Zu jedem Modell D von E gibt es dann genau einen Homomorphismus vom initialen Modell von E nach D.
  3. Durch Eigenschaft 2 ist das initiale Modell von E bis auf Isomorphie eindeutig bestimmt.

↬ [Satz 6.8, Skript S. 164]

- Über der Signatur Σ sei E eine Menge von Gleichungen. Dann gilt für eine beliebige Interpretation (D, I) über Σ:
  $$(D, I) \cong (\overline{T}^0_E, Init_E) \Leftrightarrow$$

  1. (D, I) ist Modell von E
  2. (D, I) ist termerzeugt
  3. Es gibt eine Menge $Kan \subseteq Term^0_\Sigma$, so dass gilt:
     - Zu jedem $t \in Term^0_\Sigma$ existiert $t_* \in Kan$ mit $t \leftrightarrow_E t_*$
     - Für alle $s_*, t_* \in Kan$: $I(s_*) = I(t_*) \Leftrightarrow s_* = t_*$

↬ [Lemma 6.12, Skript S. 166]

3.2 Birkhoffs Kalkül

- **Birkhoffs Kalkül B**: Axiome
  $$t \doteq t \qquad \text{für alle Terme } t$$
  Regeln:
  $$\frac{s \doteq t}{t \doteq s} \quad \text{Symmetrie}$$
  $$\frac{r \doteq s,\ s \doteq t}{r \doteq t} \quad \text{Transitivität}$$
  $$\frac{s_1 \doteq t_1, \ldots, s_n \doteq t_n}{f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n)} \quad \text{Kongruenz}$$
  $$\frac{s \doteq t}{\sigma(s) \doteq \sigma(t)} \quad \text{Substitution}$$
  für alle $n \in \mathbb{N}$, Terme $r, s, t, s_1, \ldots, s_n, t_1, \ldots, t_n$, n-stelligen Funktionssymbole f und Substitutionen σ. In üblicher Weise ist dann für eine Menge E von Gleichungen und eine Gleichung $s \doteq t$ definiert
  $$E \vdash_B s \doteq t,$$
  $s \doteq t$ ist mittels **B** ableitbar aus E.

↬ [Definition 6.14, Skript S. 168]

- $$E \vdash_B s \doteq s \doteq t \quad \Rightarrow \quad E \vdash_H s \doteq t \quad \Leftrightarrow \quad E \models s \doteq t$$
  B ist also korrekt.

↬ [Korollar 6.15, Skript S. 168]

- **Ersetzungslemma**: Es seien s, t, u, v Terme mit
$$E \vdash_B s \doteq t$$
s ist Unterterm von u, v entsteht aus u, indem man an irgendwelchen Stellen s durch t ersetzt. Dann gilt
$$E \vdash_B u \doteq v$$
$\looparrowright$ [Lemma 6.16, Skript S. 169]

- $$E \vdash_B s \doteq t \quad \Leftrightarrow \quad s \leftrightarrow_E t$$
$\looparrowright$ [Satz 6.17, Skript S. 169]

3.3 Reduktionssysteme

- Ein **Reduktionssystem** $(D, \succ)$ besteht aus einer nichtleeren Menge D und einer beliebigen, binären Relation $\succ$ auf D.
$\looparrowright$ [Definition 6.18, Skript S. 170]

- Ist $(D, \succ)$ ein Reduktionssystem, dann benutzen wir die folgenden Bezeichnungen:
  $\rightarrow$ die reflexive, transitive Hülle von $\succ$
  $\xrightarrow{+}$ die transitive Hülle von $\succ$
  $\leftrightarrow$ die reflexive, transitive, symmetrische Hülle von $\succ$
$\looparrowright$ [Definition 6.19, Skript S. 170]

- 1. Ein Reduktionssystem $(D, \succ)$ heisst **konfluent**, wenn für jedes Tripel $s, s_1, s_2 \in D$ mit $s \rightarrow s_1, s \rightarrow s_2$ ein $t \in D$ existiert mit $s_1 \rightarrow t$ und $s_2 \rightarrow t$.
  2. $(D, \succ)$ heisst **lokal konfluent**, wenn für alle $s, s_1, s_2 \in D$ mit $s \succ s_1$, $s \succ s_2$ ein $t \in D$ mit $s_1 \rightarrow t$ und $s_2 \rightarrow t$ existiert.
  3. $(D, \succ)$ heisst **noethersch** (oder **wohldefiniert** oder **terminierend**), wenn es keine unendlichen Folgen $s_0 \succ s_1 \succ \ldots \succ s_i \succ \ldots$ gibt.
  4. Ein konfluentes und noethersches Reduktionssystem heisst **kanonisch**.
  5. Ein Element $s \in D$ heisst **irreduzibel** (oder eine **Normalform**) in $(D, \succ)$, wenn kein $t \in D$ existiert mit $s \succ t$.
  6. Sei $s \in D$. Ein Element $s_0 \in D$ heisst eine **Normalform für** s in $(D, \succ)$, wenn s_0 irreduzibel ist und $s \rightarrow s_0$ gilt.

$\looparrowright$ [Definition 6.20, Skript S. 170]

- Sei $(D, \succ)$ ein kanonisches Reduktionssystem. Dann gilt:
  1. Zu jedem $s \in D$ gibt es eine eindeutige Normalform. Diese bezeichnen wir mit $irr(s)$.
  2. Für $s, t \in D$ gilt
  $$s \leftrightarrow t \quad \text{gdw} \quad irr(s) = irr(t)$$
  3. $(D, \succ)$ sei berechenbar im folgenden Sinne: Es gibt einen Algorithmus, der zu jedem $t \in D$ ein t' mit $t \succ t'$ liefert, wenn ein solches existiert, und andernfalls ausgibt "t ist irreduzibel". Dann ist die Relation $\leftrightarrow$ entscheidbar.

$\looparrowright$ [Satz 6.21, Skript S. 171]

- Für ein noethersches Reduktionssystem $(D, \succ)$ gilt das folgende Beweisprinzip der **Noetherschen Induktion**:
Es sei $X \subseteq D$, so dass für alle $a \in D$ gilt
$$\{b \mid a \succ b\} \subseteq X \quad \Rightarrow \quad a \in X.$$
Dann ist $X = D$.
$\looparrowright$ [Lemma 6.22, Skript S. 171]

- Wenn $(D, \succ)$ ein noethersches und lokal konfluentes Reduktionssystem ist, dann ist $(D, \succ)$ konfluent, d.h. kanonisch.
  $\looparrowright$ [Lemma 6.23, Skript S. 172]

3.4 Termersetzungssysteme

- Ist E eine endliche Menge von Gleichungen über der Signatur Σ, dann nennen wir das Reduktionssytem $(Term_\Sigma, \stackrel{1}{\to}_E)$ ein **Termersetzungssystem**. Da dieses durch Σ und E eindeutig bestimmt ist, sprechen wir kürzer vom Termersetzungssystem (Σ, E).
  $\looparrowright$ [Definition 6.28, Skript S. 175]
- (Σ, E) sei ein kanonisches Termersetzungssystem.
  1. Zu jedem Term t gibt es genau einen irreduziblen Term $irr(t)$ mit $t \to_E irr(t)$.
  2. Für beliebige Terme s, t gilt:
  $$E \models s \doteq t \quad \Leftrightarrow \quad irr(s) = irr(t).$$
  3. Die Gültigkeit einer Gleichung in der Theorie von E ist entscheidbar.

  $\looparrowright$ [Satz 6.30, Skript S. 175]

4 Stärkere Logiken

4.1 Prädikatenlogik zweiter Ordnung

- **Sonderzeichen** (wie in der PL1):
  $$(\;,\;) \;,\; \doteq \;,\; \neg \;,\; \wedge \;,\; \vee \;,\; \to \;,\; \leftrightarrow \;,\; \forall \;,\; \exists$$
  Variablen:

$Var =$	$Ivar \cup Mvar \cup Rvar$ (paarweise disjunkt)
$Ivar$:	unendlich viele Individuenvariablen $v_0, v_1, \dots$
	Notation: $x, y, z, \dots$
$Mvar$:	unendlich viele Mengenvariablen oder einstellige Prädikatvariablen $M_0, M_1, \dots$
	Notation: $X, Y, Z, \dots$
$Rvar$:	unendlich viele Relationenvariablen oder zweistellige Prädikatenvariablen $R_0, R_1, \dots$
	Notation $U, V, \dots$

  Signatur (wie in der PL1):
  $$\Sigma = (F_\Sigma, P_\Sigma, \alpha_\Sigma)$$
  Terme (wie in der PL1):
  $$Term_\Sigma$$
  atomare Formeln:

$s \doteq t$	für	Terme s, t
$p(t_1, \dots, t_n)$	für	$p \in P_\Sigma$, $\alpha_\Sigma(p) = n$, $t_1, \dots, t_n \in Term_\Sigma$
$X(t)$	für	Mengenvariable X und Terme t
$U(s, t)$	für	Relationenvariable U und Terme s, t

  Formeln:
  For^2_Σ (kurz For_Σ oder For) enthält genau
  - alle atomaren Formeln
  - $true$, $false$

– mit $A, B \in For^2_\Sigma$, $x \in Ivar$, $X \in Mvar$, $U \in Rvar$ auch

$$\neg A \quad , \quad (A \wedge B) \quad , \quad \ldots \quad , \quad \forall xA \quad , \quad \exists xA \quad , \quad \forall XA \quad , \quad \exists XA \quad , \quad \forall UA \quad , \quad \exists UA$$

Die Begriffe "freie Variable", "gebundene Variable", "Substitution", "kollisionsfreie Substitution", "Präfix", "Allabschluss", "Existenzabschluss" u.ä. werden entsprechend der PL1 gebildet.

↬ [Definition 7.2, Skript S. 177]

- Zu einer Interpretation (D, I) (wie in der PL1) sind jetzt neben Belegungen

$$\beta : Ivar \to D$$

auch Belegungen

$$\gamma : Mvar \to P(D) \quad \text{und} \quad \delta : Rvar \to P(D \times D)$$

zu betrachten (wobei P die Potenzmenge ist). Zu $D, I, \beta, \gamma, \delta$ ist $val_{D,I,\beta,\gamma,\delta}$ entsprechend dem Vorgehen in der PL1 definiert.
Auf Termen (wie in der PL1; hängt nicht von γ und δ ab):

$$val_{D,I,\beta,\gamma,\delta}(t) = val_{D,I,\beta}(t)$$

Auf Formeln:

$$\left.\begin{array}{l} val_{D,I,\beta,\gamma,\delta}(p(t_1, \ldots, t_n)) \\ val_{D,I,\beta,\gamma,\delta}(s \doteq t) \end{array}\right\} \text{wie in der PL1}$$

$$\begin{array}{lcl} val_{D,I,\beta,\gamma,\delta}(X(t)) = W & \Leftrightarrow & val_{D,I,\beta,\gamma,\delta}(t) \in \gamma(X) \\ val_{D,I,\beta,\gamma,\delta}(U(s,t)) = W & \Leftrightarrow & (val_{D,I,\beta,\gamma,\delta}(s), val_{D,I,\beta,\gamma,\delta}(t)) \in \delta(U) \end{array}$$

$$\left.\begin{array}{l} val_{D,I,\beta,\gamma,\delta}(\forall XA) = W \\ \neg A, (A \wedge B), (A \vee B), (A \to B), (A \leftrightarrow B), \forall xA, \exists xB \end{array}\right\} \text{wie in der PL1}$$

$$\begin{array}{lcl} val_{D,I,\beta,\gamma,\delta}(\forall XA) = W & \Leftrightarrow & \text{für jede Modifikation } \gamma^M_X \text{ von } \gamma \\ & & \text{gilt } val_{D,I,\beta,\gamma^M_X,\delta}(A) = W \\ val_{D,I,\beta,\gamma,\delta}(\forall UA) = W & \Leftrightarrow & \text{für jede Modifikation } \delta^R_U \text{ von } \delta \\ & & \text{gilt } val_{D,I,\beta,\gamma,\delta^R_U}(A) = W \end{array}$$

Existenzquantoren: Entsprechend mit "Es gibt...". (Modifikationen sind entsprechend der Begriffsbildung bei Belegungen von Individuenvariablen definiert.)

$$\begin{array}{rcl} (D, I) \text{ heisst } \textbf{Modell} \text{ von } A & :\Leftrightarrow & val_{D,I,\beta,\gamma,\delta}(A) = W \text{ für alle } \beta, \gamma, \delta. \ (D, I) \text{ ist Modell} \\ & & \text{einer Formelmenge } M \\ & :\Leftrightarrow & (D, I) \text{ ist Modell jeder Formel in } M. \\ M \models A & :\Leftrightarrow & \text{Jedes Modell von } M \text{ ist Modell von } A. \\ A \text{ allgemeingültig} & :\Leftrightarrow & \models A \text{ (d.h. } \emptyset \models A) \\ A \text{ erfüllbar} & :\Leftrightarrow & \neg A \text{ ist nicht allgemeingültig.} \\ & & \text{(D.h.: Es gibt } D, I, \beta, \gamma, \delta \text{ mit } val_{D,I,\beta,\gamma,\delta}(A) = W) \end{array}$$

↬ [Definition 7.3, Skript S. 178]

- Auf der zweiten Stufe kann es Formeln geben, in denen kein Signatursymbol und auch nicht $\doteq$ auftritt, z.B. $X(y)$. Ist eine solche Formel geschlossen, dann hängt ihr Wahrheitswert bei einer Inteetation (D, I) (nicht nur nicht von Belegungen, sondern auch) höchstens von D, nicht von I ab. Z.B. ist $\exists X \exists y X(y)$ allgemeingültig, $\forall X \exists y X(y)$ unerfüllbar, $\exists X (\exists y X(y) \wedge \exists y \neg X(y))$ gilt genau bei den (D, I) mit $\#D \geq 2$.
↬ [Bemerkung 7.4, Skript S. 179]

- In der PL2 ist die Gleichheit durch eine Formel charakterisierbar.
↬ [Satz 7.5, Skript S. 179]

- Durch das Peano'sche Axiomensystem $P1, \ldots, P5$ in der PL2 sind die natürlichen Zahlen mit Addition und Multiplikation bis auf Isomorphie eindeutig charakterisiert.
↬ [Satz 7.6, Skript S. 179]

- In der PL2 ist die Endlichkeit einer Interpretation durch eine Formel charakterisierbar.
  $\looparrowright$ [Satz 7.8, Skript S. 179]

- Die PL2 ist nicht kompakt, d.h.:
  - Aus $M \models A$ braucht i.a. nicht zu folgen:

    $$E \models A \text{ für eine endliche Teilmenge } E \text{ von } M$$

  - Es kann sein, dass jede endliche Teilmenge einer Formelmenge M ein Modell hat, diese selbst aber nicht.

  $\looparrowright$ [Satz 7.9, Skript S. 179]

- Für die PL2 kann es keinen korrekten und vollständigen Kalkül geben.
  $\looparrowright$ [Satz 7.10, Skript S. 180]

4.2 Dynamische Logik

- **Sonderzeichen** (wie in der PL1), zusätzlich:

  $$[\ ,\ ]\ ,\ <\ ,\ >\ ,\ skip\ ,\ abort\ ,\ :=\ ,\ ;\ ,\ if\ ,\ then\ ,\ else\ ,\ while\ ,\ do$$

  $$\left.\begin{array}{l} Variable,\ Var \\ Signatur,\ \Sigma \\ Terme,\ Term_\Sigma \\ atomare\ Formeln \end{array}\right\} \text{wie in der PL1}$$

  Boole'sche Ausdrücke, $Bexp_\Sigma$, sind die quantorenfreien PL1-Formeln über Σ. Sie werden aus den atomaren Formeln und $true$, $false$ durch die aussagenlogischen Operatoren gebildet.
  Programme, $Prog_\Sigma$:

  - $skip$ und $abort$ sind Programme.
  - Mit $x \in Var$ und $t \in Term_\Sigma$ ist $(x := t)$ ein Programm (Zuweisung).
  - Sind π, ρ Programme und ist ϵ ein Boole'scher Ausdruck, dann sind auch Programme:

    $$\begin{array}{ll} (\pi;\ rho) & \text{(Hintereinanderausführung)} \\ if\ \epsilon\ then\ \pi\ else\ \rho & \text{(Verzweigung)} \\ while\ \epsilon\ do\ \pi & \text{(Schleife)} \end{array}$$

  (Man beachte, dass wegen der Klammerung der Hintereinanderausführung eine schliessende Klammer für Verzweigung bzw. Schleife unnötig ist.)
  Formeln, For_Σ:

  - Alle Boole'schen Ausdrücke sind Formeln.
  - Mit $A, B \in For_\Sigma$, $x \in Var$, $\pi \in Prog_\Sigma$ sind auch Formeln:

    $$\neg A \quad , \quad (A \wedge B) \quad , \quad \ldots \quad , \quad [\pi]A \quad , \quad \langle\pi\rangle A$$

  $\looparrowright$ [Definition 7.13, Skript S. 180]

- Zu einer Interpretation (D, I) über Σ und einem Zustand β, d.h. einer Belegung der Variablen, sind $val_{D,I,\beta}(t)$ für Terme t und $val_{D,I,\beta}(\epsilon)$ für Boole'sche Ausdrücke ϵ definiert wie in der PL1.
  Es sei $Sta := (Var \to D)$ die Menge der Zustände. Die **Semantik der Programme** wird festgelegt durch die Definition einer Relation $\rho(\pi) \subseteq Sta \times Sta$ für jedes $\pi \in Prog_\Sigma$. ($\rho(\ldots)$ hängt von (D, I) ab,

aus Gründen der einfacheren Notation schreiben wir aber einfach $\rho(\pi)$ statt $\rho(\pi)_{D,I}$.)
$\rho(abort)$ ist die leere Relation

$$\begin{array}{rcl}
\beta\rho(skip)\gamma & :\Leftrightarrow & \beta = \gamma \\
\beta\rho(x := t)\gamma & :\Leftrightarrow & \gamma = \beta_x^{val_{D,I,\beta}(t)} \\
\beta\rho(\pi_1; \pi_2)\gamma & :\Leftrightarrow & \text{Es gibt } \delta \in Sta \text{ mit } \beta\rho(\pi_1)\delta \text{ und } \delta\rho(\pi_2)\gamma \\
\beta\rho(if\ \epsilon\ then\ \pi_1\ else\ \pi_2)\gamma & :\Leftrightarrow & (val_{D,I,\beta}(\epsilon) = W \text{ und } \beta\rho(\pi_1)\gamma) \text{ oder} \\
& & (val_{D,I,\beta}(\epsilon) = F \text{ und } \beta\rho(\pi_2)\gamma) \\
\beta\rho(while\ \epsilon\ do\ \pi)\gamma & :\Leftrightarrow & \text{Es gibt } n \in \mathbb{N} \text{ und Zustände } \beta_0, \ldots, \beta_n \text{ mit:} \\
& & -\ \beta_0 = \beta,\ \beta_n = \gamma \\
& & -\ \forall i \text{ mit } 0 \leq i < n\ :\ val_{D,I,\beta}(\epsilon) = W \text{ und } \beta_i\rho(\pi)\beta_{i+1} \\
& & -\ val_{D,I,\beta_n}(\epsilon) = F
\end{array}$$

Zu gegebenem π und β gibt es stets höchstens ein γ mit $\beta\rho(\pi)\gamma$: Die Relation $\rho(\pi)$ ist rechtseindeutig und hätte auch als partielle Funktion formuliert werden können. Das würde sich ändern, wenn man auch indeterministische Programme zulässt.
$\beta\rho(\pi)\gamma$ besagt, dass π, beginnend im Zustand β, terminiert und zum Zustand γ führt. π, beginnend in β, terminiert genau dann, wenn es ein solches γ gibt. Etwa terminiert *abort* in keinem Zustand, ebenso *while true do* ρ (für beliebiges ρ). Das Programm *while* ϵ *do skip* terminiert genau bei Beginn in Zuständen β mit $val_{D,I,\beta}(\epsilon) = F$ (und dann nach 0 Schritten).
Formeln:
Für Boole'sche Ausdrücke ϵ ist nach Annahme $val_{D,I,\beta}$ für Formeln

$$\neg A \quad , \quad (A \wedge B) \quad , \quad (A \vee B) \quad , \quad (A \rightarrow B) \quad , \quad (A \leftrightarrow B) \quad , \quad \forall x A \quad , \quad \exists x A$$

definiert entsprechend dem Vorgehen in der PL1.

$$\begin{array}{rcl}
val_{D,I,\beta}([\pi]A) = W & :\Leftrightarrow & \text{Für jedes } \gamma \text{ mit } \beta\rho(\pi)\gamma \text{ gilt } val_{D,I,\gamma}(A) = W \\
val_{D,I,\beta}(\langle\pi\rangle A) = W & :\Leftrightarrow & \text{Es gibt ein } \gamma \text{ mit } \beta\rho(\pi)\gamma \text{ und } val_{D,I,\gamma}(A) = W
\end{array}$$

Man bemerke, dass (in unserem Fall von deterministischen Programmen) gilt: $val_{D,I,\beta}([\pi]A) = W$ ganau dann, wenn: Falls ein γ existiert mit $\beta\rho(\pi)\gamma$, dann ist für dieses (dann eindeutig bestimmte) γ $val_{D,I,\gamma}(A) = W$.
(D, I) heisst **Modell** von A, wenn $val_{D,I,\beta}(A) = W$ für alle Zustände β. Entsprechend sind wie in der PL1 erklärt: Modell einer Formelmenge M, allgemeingültig, erfüllbar, sowie die Folgerkeitsbeziehung $\models$.
$\looparrowright$ [Definition 7.14, Skript S. 181]

- Die natürlichen Zahlen mit Addition und Multiplikation lassen sich in der DL bis auf Isomorphie eindeutig charakterisieren.
  $\looparrowright$ [Satz 7.16, Skript S. 183]

- Die DL ist nicht kompakt.
  $\looparrowright$ [Satz 7.17, Skript S. 183]

- Für die DL gibt es keinen korrekten und vollständigen Kalkül.
  $\looparrowright$ [Korollar 7.18, Skript S. 184]

- Zu $\epsilon \in Bexp_\Sigma$, $\pi \in Prog_\Sigma$, $A \in For_\Sigma$ definieren wir für alle $n \in \mathbb{N}$ Formeln $Loop_n(\epsilon, \pi, A)$ wie folgt:

$$\begin{array}{rcl}
Loop_0(\epsilon, \pi, A) & = & \neg\epsilon \rightarrow A \\
Loop_{n+1}(\epsilon, \pi, A) & = & [\pi]Loop_n(\epsilon, \pi, A)
\end{array}$$

  $\looparrowright$ [Definition 7.20, Skript S. 184]

- Zu ϵ, π, A sowie einer Interpretation (D, I) und einem Zustand β sei $\beta_0, \beta_1, \ldots$ die eindeutig bestimmte Folge von Zuständen mit $\beta_0 = \beta, \beta_i \rho(\pi) \beta_{i+1}$. Dann gitl für jedes n:

$$val_{D,I,\beta}(Loop_n(\epsilon, \pi, A)) = W \quad \Leftrightarrow \quad \text{Wenn } val_{D,I,\beta_i}(\epsilon) = W \text{ für alle } i < n \text{ und } val_{D,I,\beta_n}(\epsilon) = F, \text{ dann } val_{D,I,\beta_n}(A) = W.$$

↬ [Korollar 7.21, Skript S. 184]

- Für beliebige D, I, β gilt

$$val_{D,I,\beta}([while\ \epsilon\ do\ \pi]A) = W \quad \Leftrightarrow \quad \text{Für alle } n \in \mathbb{N} : val_{D,I,\beta}(Loop_n(\epsilon, \pi, A)) = W$$

↬ [Lemma 7.22, Skript S. 185]

5 Omega Automaten

- Sei V ein (weiterhin endliches) Alphabet. Mit V^ω bezeichnen wir die Menge der **unendlich langen Wörtern** mit Buchstaben aus V. Für $n \in \mathbb{N}$ bezeichnet $w(n)$ den n-ten Buchstaben in w und $w \downarrow (n)$ das endliche Anfangstück $w(0) \ldots w(n)$ von w.
Wir nennen ein Wort $w \in V^\omega$ manchmal auch ein ω-Wort über V.
↬ [Definition 9.15, Skript S. 211]

- **Operationen**

1. Ist $K \subseteq V^*$ eine Menge (endlicher) Wörter, so bezeichnet K^ω die Menge der unendlichen Wörter der Form
$$w_1 \ldots w_i \ldots \text{ mit } w_i \in K \text{ für alle } i$$
2. Die Verkettung zweier unendlich langer Wörter macht keinen Sinn, es ist jedoch möglich einem unendlich langen Wort ein endliches vorzuschalten. Ist $K \subseteq V^*$ und $J \subseteq V^\omega$, dann setzen wir:
$$KJ = \{w_1 w_2 \mid w_1 \in K, w_2 \in J\}$$
3. Ist $K \subseteq V^*$ eine Menge endlicher Wörter, dann ist
$$\overrightarrow{K} = \{w \in V^\omega \mid w \downarrow (n) \in K \text{ für unendlich viele } n\}$$
Manche Autoren benutzen $\lim(K)$ anstelle von $\overrightarrow{K}$.

↬ [Definition 9.16, Skript S. 211]

- Sei $A = (S, V, s_0, \delta, F)$ ein nicht deterministischer endlicher Automat. Für ein ω-Wort $w \in V^\omega$ nennen wir eine Folge $s_1, \ldots, s_n, \ldots$ eine **Berechnungsfolge** (englisch: run) für w, wenn für alle $0 \leq m$ gilt

$$s_{n+1} \in \delta(s_n, w(n))$$

Die von A akzeptierte ω-Sprache wird definiert durch

$$L^\omega(A) = \left\{ w \in V^\omega \;\middle|\; \begin{array}{l} \text{es gibt eine Berechnungsfolge für } w \\ \text{mit unendlich vielen Finalzuständen} \end{array} \right\}$$

Wir nennen eine Menge L von ω-Wörtern **ω-regulär**, wenn es einen endlichen Automaten A gibt mit $L^\omega(A) = L$.
Gelegentlich braucht man auch den Begriff einer Berechnungsfolge unabhängig davon, welches Wort damit assoziiert ist.
Eine Berechnungsfolge für den Automaten $A = (S, V, s_0, \delta, F)$ ist eine Folge $s_1, \ldots, s_n, \ldots$ von Zuständen, so dass für alle $0 \leq n$ ein $a \in V$ existiert mit $s_{n+1} \in \delta(s_n, a)$. Eine Berechnungsfolge heisst akzeptierend, wenn unendlich viele der s_i Endzustände sind.
Man beachte, dass eine Berechnungsfolge mit s_1 beginnt.
Ein **Büchi Automat** A unterscheidet sich also in nichts von einem endlichen Automaten, nur die Definition, wann A ein ω-Wort akzeptiert ist neu.
↬ [Definition 9.17, Skript S. 211]

- Die Frage, ob für einen Büchi Automaten B die Menge der akzeptierten Wörter nicht leer ist, d.h. $L^\omega(B) \neq \emptyset$, ist entscheidbar.
  $\looparrowright$ [Satz 9.18, Skript S. 212]

- Sei A ein endlicher Automat und $K = L(A)$. Dann gilt
  1. $L^\omega(A) \subseteq \overrightarrow{K}$
  2. Falls A deterministisch ist, gilt sogar $L^\omega(A) = \overrightarrow{K}$

  $\looparrowright$ [Lemma 9.19, Skript S. 213]

- Eine Sprache $L \subseteq V^\omega$ wird von einem deterministischen Büchi Automaten akzeptiert, gdw. es eine reguläre Sprache $K \subseteq K^*$ gibt mit $L = \overrightarrow{K}$.
  $\looparrowright$ [Korollar 9.20, Skript S. 213]

- Es gibt Sprachen $L \subseteq V^\omega$, die von einem nicht deterministischen Büchi Automaten akzeptiert werden, aber von keinem deterministischen.
  $\looparrowright$ [Korollar 9.21, Skript S. 213]

- Sind L_1, L_2 ω-reguläre Sprachen und ist K eine reguläre Sprache, dann ist auch
  1. $L_1 \cup L_2$ ω-regulär,
  2. K^ω ω-regulär, falls $\varepsilon \notin K$,
  3. KL_1 ω-regulär,
  4. $V^\omega \backslash L_1$ ω-regulär,
  5. $L_1 \cap L_2$ ω-regulär.

  $\looparrowright$ [Satz 9.22, Skript S. 214]

- Eine Teilmenge $L \subseteq V^\omega$ ist eine ω-reguläre Menge von ω-Wörtern, genau dann, wenn L eine endliche Vereinigung von Mengen der
  $$JK^\omega$$
  für reguläre Mengen $J, K \subseteq V^*$ ist, wobei $\varepsilon \notin K$.
  $\looparrowright$ [Satz 9.23, Skript S. 215]

- Sei $A = (Q, V, s_0, \delta, F)$ ein Büchi Automat. Dann definieren wir für $p, q \in Q$ und $u, w \in V^*$
  1. $L_{p,q} = \{w \in V^* \mid q \in \delta(p, w)\}$
  2. $L^F_{p,q} = \left\{ w = a_0 \dots a_k \in V^* \;\middle|\; \begin{array}{llll} \multicolumn{4}{l}{\text{es gibt eine Folge von Zuständen } q_0 \dots q_k \text{ mit}} \\ q_0 & = & p & \\ q_k & = & q & \\ q_{i+1} & \in & \delta(q_i, a_i) & \text{für alle } 0 \leq i < k \\ q_i & \in & F & \text{für ein } 0 \leq i \leq k \end{array} \right\}$
  3. $u \equiv_A$ gdw für alle $p, q \in Q$ gilt
  $$u \in L_{p,q} \Leftrightarrow w \in L_{p,q} \quad \text{und} \quad u \in L^F_{p,q} \Leftrightarrow w \in L^F_{p,q}$$

  $\looparrowright$ [Definition 9.24, Skript S. 216]

- Die Äquivalenzrelation $\equiv_A$ hat folgende Eigenschaften:
  1. für alle $p, q \in Q$ sind $L^F_{p,q}$ und $L_{p,q}$ reguläre Mengen,
  2. für $u \in V^\omega$ gilt für die Äquivalenzklasse M_w
  $$M_w = \bigcap_{(p,q)\in P} L_{p,q} \cap \bigcap_{(p,q)\notin P} \sim L_{p,q} \cap \bigcap_{(p,q)\in R} L^F_{p,q} \cap \bigcap_{(p,q)\notin R} \sim L^F_{p,q}$$
  wobei $P = \{(p,q) \in Q^2 \mid w \in L_{p,q}\}$ und $R = \{(p,q) \in Q^2 \mid w \in L^F_{p,q}\}$

3. $\equiv_A$ besitzt endlich viele Äquivalenzklassen
4. jede Äquivalenzklasse von $\equiv_A$ ist eine reguläre Menge
5. sind U, V Äquivalenzklassen von $\equiv_A$, dann folgt aus $UV^\omega \cap L^\omega(A) \neq \emptyset$ schon $UV^\omega \subseteq L^\omega(A)$

$\looparrowright$ [Lemma 9.25, Skript S. 216]

- **Satz von Ramsey**: Sei $P_1, \ldots, P_k$ eine endliche Partition aller zweielementigen Teilmengen der Menge $\mathbb{N}$ der natürlichen Zahlen. Dann gibt es eine unendliche Teilmenge $T \subseteq \mathbb{N}$ und ein i mit $1 \leq i \leq k$, so dass alle zweielementigen Teilmengen von Elementen aus T in P_i liegen. In Formeln:
Aus $P_1 \uplus \ldots \uplus P_k = \mathbb{N}^{[2]}$ folgt die Existenz einer unendlichen Menge $T \subseteq \mathbb{N}$, so dass $T^{[2]} \subseteq P_i$ für ein i.
$\looparrowright$ [Satz 9.26, Skript S. 217]

- Zu jedem ω-Wort $w \in V^\omega$ gibt es eine $\equiv_A$-Äquivalenzrelation U und V mit $w \in UV^\omega$.
$\looparrowright$ [Lemma 9.27, Skript S. 217]

- Ist $L \subseteq V^\omega$ eine ω-reguläre Menge, dann auch $V^\omega \backslash L$.
$\looparrowright$ [Satz 9.28, Skript S. 218]

- Zu jedem Büchi Automaten $C = (S, V, S_0, \delta, F)$ mit einer Menge von Anfangszuständen gibt es einen Büchi Automaten A mit einem einzigen Anfangszustand und

$$L^\omega(C) = L^\omega(A)$$

$\looparrowright$ [Lemma 9.29, Skript S. 218]

- Ein **erweiterter Büchi Automat**

$$A = (S, V, s_0, \delta, F_1, \ldots, F_n)$$

unterscheidet sich von einem (normalen) Büchi Automaten nur dadurch, dass er statt einer einzigen Menge F von Finalzuständen endlich vieler solcher Mengen $F_1, \ldots, F_n$ enthält.
Eine Berechnungsfolge w wird akzeptiert, wenn es eine Berechnungsfolge s für w gibt, die für jedes j, $1 \leq j \leq n$ unendlich viele Zustände aus F_j enthält. Die von A akzeptierte ω-Sprache kann dann definiert werden als:

$$L^\omega(A) = \left\{ w \in V^\omega \;\middle|\; \begin{array}{l} \text{es gibt eine Berechnungsfolge } s \text{ für } w, \\ \text{so dass für jedes } j,\ 1 \leq j \leq n,\ \text{die Menge } \{i \mid s_i \in F_j\} \text{ unendlich ist.} \end{array} \right\}$$

$\looparrowright$ [Definition 9.30, Skript S. 218]

- Zu jedem erweiterten Büchi Automaten A_e gibt es einen einfachen Büchi Automaten A mit

$$L^\omega(A_e) = L^\omega(A)$$

$\looparrowright$ [Lemma 9.31, Skript S. 219]

- Ein **Müller Automat** $M = (S, V, s_0, \delta, \mathcal{F})$ ist ein endlicher Automat $M = (S, V, s_0, \delta)$ ohne Ausgabe von Endzuständen, aber stattdessen mit einer Menge F von Endzustandsmengen, d.h. für alle $F \in \mathcal{F}$ gilt $F \subseteq S$.
Ist $\sigma = s_1, \ldots, s_n, \ldots$ eine Zustandsfolge, so bezeichnet $In(\sigma)$ die Menge der Zustände, die unendlich oft in σ vorkommen, also

$$In(\sigma) = \{s \in S \mid \text{es gibt unendlich viele } n \text{ mit } s_n = s\}$$

Die von einem Müller Automat $M = (S, V, s_0, \delta, \mathcal{F})$ akzeptierte ω-Sprache wird efiniert durch

$$L^\omega(M) = \{w \in V^\omega \mid In(\sigma) \in \mathcal{F} \text{ für eine Berechnungsfolge } \sigma \text{ für } w\}$$

$\looparrowright$ [Definition 9.32, Skript S. 219]

- Die Klasse der von nicht deterministischen Büchi Automaten akzeptierten ω-Sprachen stimmt überein mit der von nicht deterministischen Müller Automaten akzeptierten ω-Sprachen.
$\looparrowright$ [Lemma 9.33, Skript S. 219]

6 Temporale Logiken

6.1 Lineare Temporale Logik

- Sei Σ eine Menge aussagenlogischer Atome. Die Menge $LTLFor$ (bzw. genauer $LTLFor_\Sigma$ falls erforderlich) der **LTL-Formeln** ist definiert durch

  1. $\Sigma \subseteq LTLFor$
  2. $1, 0 \in LTLFor$
  3. Liegen A, B in $LTLFor$, dann auch alle aussagenlogischen Kombinationen von A und B.
  4. Für $A, B \in LTLFor$ gilt auch
     (a) $\Box A \in LTLFor$ und
     (b) $\Diamond B \in LTLFor$ und
     (c) $A \mathbf{\ U\ } B \in LTLFor$ und
     (d) $\mathbf{X}\,A$

  Die Symbole $\Box$, $\Diamond$, $\mathbf{X}$ und $\mathbf{U}$ heissen temporale Modaloperatoren. Die Notation für die temporalen Modaloperatoren ist leider nicht einheitlich. Für die Semantik von LTL benutzen wir omega-Strukturen.
  ↬ [Definition 10.1, Skript S. 225]

- Eine **omega-Struktur** $R = (\mathbb{N}, <, \xi)$ für eine aussagenlogische Signatur P besteht aus der geordneten Menge der natürlichen Zahlen

  $$(\mathbb{N}, <)$$

  interpretiert als Menge abstrakter Zeitpunkte und einer Funktion

  $$\xi : \mathbb{N} \to 2^P$$

  mit der Interpretation

  $$p \in \xi(n) \quad \Leftrightarrow \quad \text{in } R \text{ ist } p \text{ zum Zeitpunkt } n \text{ wahr}$$

  Für $\xi : \mathbb{N} \to 2^P$ und $n \in \mathbb{N}$ steht ξ_n für das bei n beginnende Endstück von ξ. In Zeichen

  $$\xi_n(m) = \xi(n + m)$$

  Insbesondere gilt $\xi_0 = \xi$.
  ↬ [Definition 10.2, Skript S. 225]

- Sei $R = (\mathbb{N}, <, \xi)$ eine omega-Struktur und A eine TLT Formel.

$\xi \models p$	gdw	$p \in \xi(0)$ (p ein AL Atom)
$\xi \models op(A, B)$		für aussagenlogische Kombinationen $op(A, B)$ von A und B wie üblich
$\xi \models \Box A$	gdw	für alle $n \in \mathbb{N}$ gilt $\xi_n \models A$
$\xi \models \Diamond A$	gdw	es gibt ein $n \in \mathbb{N}$ mit $\xi_n \models A$
$\xi \models A \mathbf{\ U\ } B$	gdw	es gibt $n \in \mathbb{N}$ mit $\xi_n \models B$ und für alle m mit $0 \leq m < n$ gilt $\xi_m \models A$
$\xi \models \mathbf{X}\,A$	gdw	$\xi_1 \models A$

  ↬ [Definition 10.3, Skript S. 226]

- Die folgenden Äquivalenzen gelten in allen Zeitstrukturen:

  $$\begin{aligned} \Diamond A &\leftrightarrow 1 \mathbf{\ U\ } A \\ \Box A &\leftrightarrow \neg(1 \mathbf{\ U\ } \neg A) \end{aligned}$$

  ↬ [Lemma 10.4, Skript S. 226]

- **Zusätzliche Operatoren**:

$\xi \models A \ \mathbf{U}_w B$	gdw	für alle $n \in \mathbb{N}$ gilt $\xi_n \models (A \wedge \neg B)$ oder es gibt $n \in \mathbb{N}$ mit $\xi_n \models B$ und für alle m mit $0 \leq m < n$ gilt $\xi_m \models A$
$\xi \models A \mathbf{V} B$	gdw	$\xi \models B$ und für alle $n \in \mathbb{N}$ gilt falls $\xi_n \models \neg B$ dann gibt es ein m mit $0 \leq m < n$ und $\xi_m \models A$

↬ [Definition 10.5, Skript S. 227]

- Für alle omega-Strukturen gilt:

  1. $A \ \mathbf{U} \ B \leftrightarrow A \ \mathbf{U}_w B \wedge \Diamond B$
  2. $A \ \mathbf{U}_w B \leftrightarrow A \ \mathbf{U} \ B \vee \Box(A \wedge \neg B)$
  3. $A \mathbf{V} B \leftrightarrow \neg(\neg A \ \mathbf{U} \ \neg B)$
  4. $A \ \mathbf{U} \ B \leftrightarrow (B \vee (A \wedge \mathbf{X}(A \ \mathbf{U} \ B)))$
  5. $A \mathbf{V} B \leftrightarrow (B \wedge A) \vee (B \wedge \mathbf{X}(A \mathbf{V} B))$

↬ [Lemma 10.6, Skript S. 227]

Index

V^ω, 30
sh-Formel, 5
    normierte, 5
sh-Graph
    reduzierter, 5

Abkömmling, 6
Ableitung, 6
Allabschluss, $Cl_\forall$, 10
atomare Formeln, 9

Beseitigen von Existenzquantoren, 15

C-Regel, 18

Endlichkeitssatz, 21
Ersetzungslemma, 25
Ersetzungstheorem, 15
Existenzabschluss, $Cl_\exists$, 10

Folgerbarkeit, 13
Formel
    allgemeingültig, 4
    erfüllbar, 4
    geschlossene, 10
    logisch äquivalent, 4
    Substitutionslemma, 13
Freie Algebra, 23
Funktionssymbole, 9

Grundinstanz, 16
Grundliteral, 16
Grundsubstitution, 10
Grundterm, 9

Herbrand
    Algebra, 15
    Interpretation, 15
    Modell, 16
    Satz von, 16
Horn-Formel, 5
    definite, 5

Individuenvariablen, 9
Initiale Algebra, 23
Interpretation, 4, 12
    sortierte, 21

Kalkül, 6
Klausel, 19
Koinzidenzlemma, 13
Kompaktheit der PL1, 21
Konstantensymbol, 9
Krom-Formel, 5

Literal, 16, 19
logisch äquivalent, 14
LTL-Formeln, 33

Modell, 4, 13
Modifikation, 12

Negationsnormalform, 15
Noethersche Induktion, 25

Omega Automaten
    ω-regulär, 30
    Büchi Automat, 30
        erweiterter, 32
    Berechnungsfolge, 30
    Müller Automat, 32
    Operationen, 30
omega-Struktur, 33

Prädikatenlogik
    Formel, 9
        allgemeingültig, 14
        bereinigte, 15
        erfüllbar, 14
        Matrix, 15
    logische Zeichen, 9
Prädikatssymbole, 9
Pränex-Normalform, 15

Reduktionssystem, 25
    irreduzibel, 25
    kanonisches, 25
    konfluentes, 25
        lokal, 25
    noethersch, 25
Resolution
    aussagenlogische, 6
    prädikatenlogische, 19
Robbins Algebra, 22

Satz von Ramsey, 32
Sequenz, 7, 20
    Antezedent, 7
    Kalkül, 8
    Sukzedent, 7
Signatur, 9
    ordnungssortierte, 21
Skolem-Normalformel, 15
Sonstige Kalküle
    1-Resolution, 8
    Birkhoffs Kalkül, 24
    Davis-Putnam-Loveland Verfahren, 8
    Hilbertkalkül, 19
    Numerische Verfahren, 8

Stelligkeit, 9
Substitution, 10
    kollisionsfreie, 11
    Komposition, 11

Tableau, 17
    G_l-Regel, 18
    G_r-Regel, 18
    V-Regel, 18
    α-Regel, 7
    β-Regel, 7
    δ-Regel, 17
    γ-Regel, 17
    aussagenlogisches, 6
    erschöpftes, 7
    geschlossenes, 7
    Modell, 19
    prädikatenlogisches, 18
    Voraussetzungsregel, 7
Tautologie, 5, 14
Teilterm, 9
Term, 9
    sortierter, 21
    Substitutionslemma, 13
Termersetzungssystem, 26

Unifikation, 11
Unifikationsalgorithmus, 12
Unifikator, 11
    allgemeinster, 11
Unterterm, 9

Variable
    freie, 10
    gebundene, 10
Variablenbelegung, 12
Variablenumbenennung, 11
Variante, 19
Vorzeichenformel, 6
    Typen, 6

Wirkungsbereich, 10