
Logik und Semantik
1. Vorlesung
Einführung

Carola Eschenbach
– WSV –

Matthias Jantzen
– TGI –

Grundlagenveranstaltung des (Diplom-) Hauptstudiums
Bereich Theoretische Informatik

Heutige Sitzung

Thema und Ziele der Vorlesung
Organisatorisches
Einstieg Logik

Logik & Semantik: Spezifikation & Verifikation

Das Ziel: **Korrektheit von Systemen**

- Die Gründe
 - Sicherheit
 - Kosten: Revision von Systemen ist teuer!

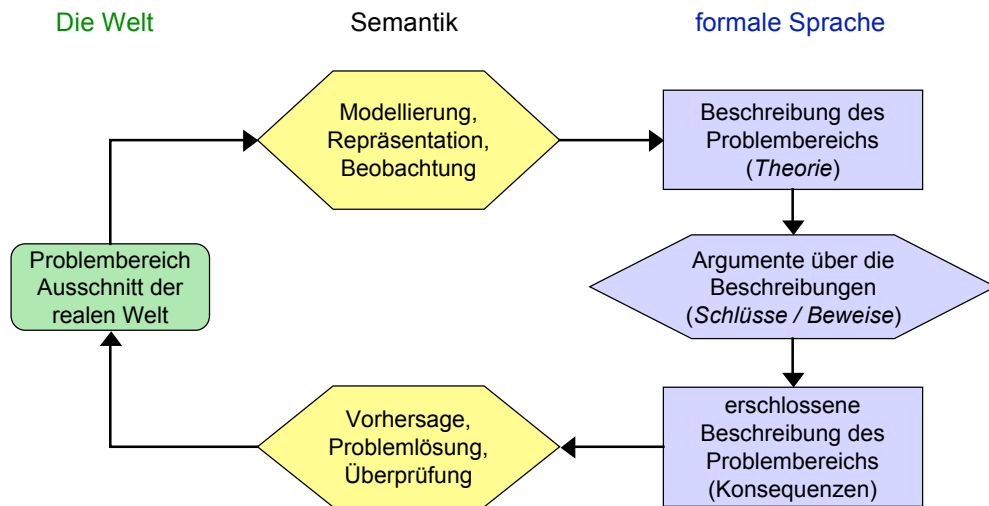
Der Weg: **Modellierung und Analyse von Systemen**

- Modellierung: Spezifikation durch sprachliche Beschreibung
- Rationale Analyse: Nachweis (Verifikation), dass das System die Bedingungen der Spezifikation erfüllt.

Was wird spezifiziert und verifiziert?

- Datenstrukturen
 - Programme
 - Hardware
- } gegebenenfalls integriert / eingebettet

Die deduktive Methode



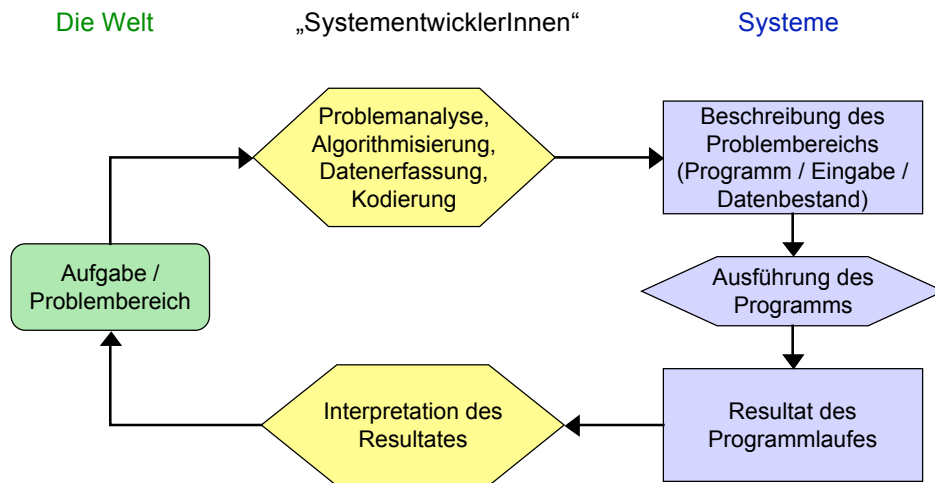
Ch. Habel / C. Eschenbach: Logik & Semantik

1 – 4

Deduktive Methode

- Methodische Grundlage der „exakten“ Wissenschaften.
 - Am deutlichsten ist diese Methode erkennbar in den Naturwissenschaften, etwa der Physik:
Wissenschaftliche Theorien werden in mathematischer Formulierung ausgearbeitet. Mathematisch fundierte Schlüsse führen zu Annahmen, die sich aus der Theorie ableiten lassen. Diese Konsequenzen werden dann durch Experiment und Beobachtung in der Realität überprüft.
 - Das zentrale Grundprinzip der deduktiven Methode ist der Zyklus von Hypothesengenerierung und Hypothesenüberprüfung:
 - Beobachtung / Analyse des Gegenstandsbereichs führt zu Theorien, die als Hypothesen anzusehen sind. Diese Hypothesen sollten natürlich durch die Realität, d.h. durch Beobachtungen, gestützt sein.
 - Die Konsequenzen der Theorie, d.h. aus der Theorie abgeleitete Sätze, stellen neue Aussagen über den Gegenstandsbereich dar, die nun wiederum an der Realität überprüft werden sollen.
 - Die Überprüfung sollte anstreben, die Theorie zu falsifizieren, d.h. Bedingungen zu finden, unter denen die Theorie falsche Vorhersagen macht. Ist es unmöglich, die Theorie zu falsifizieren, dann ist die Theorie bestätigt. Im allgemeinen ist es aber nicht leicht, diese ‚Unmöglichkeit‘ nachzuweisen.

Systementwicklung in der Informatik



Ch. Habel / C. Eschenbach: Logik & Semantik

1 – 5

- Fortsetzung Deduktive Methode
- Anwendung von Theorien: Erfolgreiche, d.h. gut bestätigte Theorien, können für die Vorhersage der Realität verwendet werden. Einschränkungen in der Vorhersagequalität können in den Theorien aber auch in den Schlussverfahren liegen. Beispiele: Meteorologie, Geologie (Erdbebenvorhersage), Ingenieurwissenschaften

Logik in der Informatik

Logik als Werkzeug

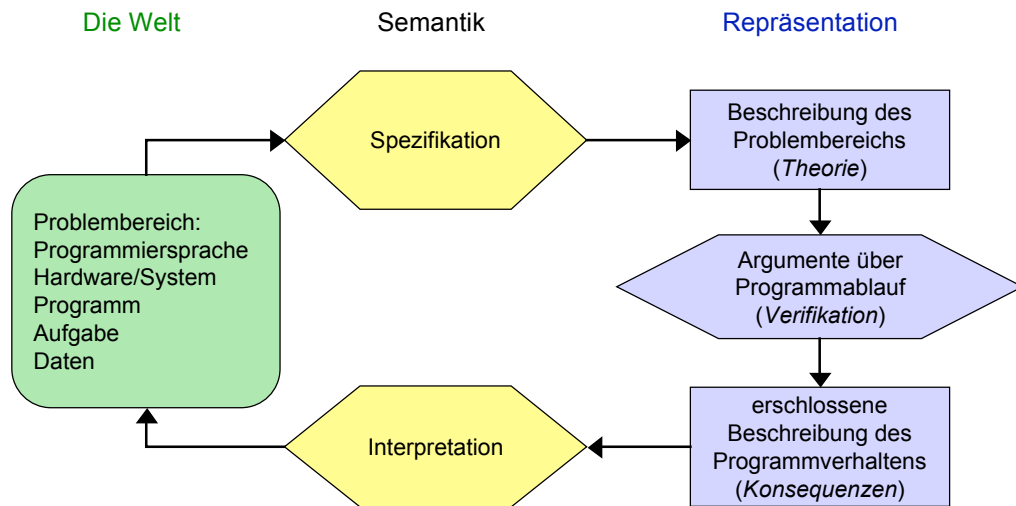
- Formale Theorien der **Spezifikation** und **Korrektheit**
 - Verifikation von Software & Hardware
 - Speziell: Parallelität und Nebenläufigkeit
- Grundlage von Berechnungen
 - Theorembeweisen/Deduktionssysteme: **Reasoning Services**
 - **Logik-Programmierung**: PROLOG

Logik als Beispiel einer formalen Sprache

zur Beschreibung / Repräsentation der Welt

- Künstliche Intelligenz, **Wissensrepräsentation und Wissensverarbeitung**, Bedeutung von Sprache und von Bildern
- **Datenbanken**, Formulierung und Prüfung von Integritätsbedingungen
- Informationssysteme und WWW: **Semantic Web, Ontologien**

Spezifikation und Verifikation – 1



Ch. Habel / C. Eschenbach: Logik & Semantik

1 – 7

Natürlich ist die oben angedeutete Aufgabe viel zu komplex, als dass sie in einem Schritt lösbar ist.

Will man ein Programm, das Quadratwurzeln berechnen soll, verifizieren, so sollte dies auch unabhängig von einer konkreten Hardware erfolgen können.

Deshalb ist auch hier eine Problemzerlegung anzuraten. So können Hardware und Programmiersprache unabhängig von dem spezifischen Programm spezifiziert werden und auf Basis dieser Spezifikationen dann Aussagen über Programmfragmente (Schleifen, Prozeduren) verifiziert werden, die bei einer Verifikation eines spezifischen Programms genutzt werden können.

Das schöne an logischen / formalen (monotonen) Verfahren ist ja, dass einmal geführte Beweise niemals ungültig werden können.

Spezifikation und Verifikation - 2

Daten

- Beschreiben die Daten(strukturen) die Welt so, wie sie ist?

Programme

- Haben die Programme das geforderte Berechnungsverhalten?

Hardware

- Hat die Hardware das geforderte Berechnungsverhalten?

Beispiel

- Berechnet das Programm **sq** auf der Hardware **IP** die Approximation der **Quadratwurzel** mit der geforderten Genauigkeit?

Verifikation

- Beweis, dass das Verhalten den Anforderungen entspricht.

Spezifikation und Verifikation - 3

Der methodische Rahmen

Sprachen – für die Spezifikation

- Logische Sprachen
 - Beschreibung der Domänen
 - Beschreibung von Berechnungen / Prozessen

Verfahren – für die Verifikation

- Beweisverfahren / Ableitungsverfahren
-

Logik & Semantik

- Semantik: sprachliche Ausdrücke \leftrightarrow Welt

Semantik & Programmiersprachen

- Einsatz des logischen Methodeninventars für die Beschreibung und Verifikation von Berechnungsprozessen

- Verifikation & Kontrolle
 - Es ist vergleichsweise einfach, zu prüfen, ob etwas ein formaler Beweis einer Aussage ist.
 - Liegt ein Beweis vor, der Eigenschaften von Software nachweist, dann weiß der Nutzer, worauf er sich verlassen kann.
 - Der Erzeuger des Programms kann für das Vertrauen des Nutzers sorgen, indem er die erforderlichen Eigenschaften beweist.
 - Der Nutzer muss aber mit dem Beweis auch etwas anfangen können (und das kann er, wenn er z.B. ein Programm hat, das die Korrektheit verlässlich prüfen kann.)

Syntax und Semantik

Syntax

- beschreibt die erlaubte Form einer Spezifikation

Syntaxis (griech.)

- Zusammenordnung, Lehre vom Satzbau z.B. wohlgeformte Formeln, Ausdrücke, Programme.

Semantik

- ist die Lehre von Bedeutungsinhalten der syntaktischen Konstrukte.

Sem (griech.)

- eines von mehreren Bedeutungselementen, Merkmalen, die zusammen ein Semen ausmachen.

Semantikansatz Prinzip Abstrahierungsvorgang

- **Übersetzersemantik**

Anweisung wird in bekannter Sprache erklärt

- **kaum Abstraktion**

- **Operationale Semantik**

Anweisung ändert Zustand einer abstrakten Maschine (vergl. Java virtual machine)

- **Verzicht auf bek. Sprache, dafür aber Maschine**

- **Denotationale Semantik**

Programm P ist Beschreibung einer Funktion f_P über dem cartesischen Produkt der Wertebereiche aller Variablen in P

- **Abstraktion von Maschine**

- **Axiomatische Semantik**

Anweisung setzt Vor- und Nachbedingungen in Beziehung

- **Abstrahierung von konkreten Zuständen**

Voraussetzungen aus Grundstudium: Klassische Logik

Aussagenlogik

- Syntax, Semantik
- Normalformen, Umformungsregeln
- Ableitungsverfahren (formale Beweise)
- Hornformeln, Resolutionsverfahren

Prädikatenlogik der ersten Stufe

- Syntax, Semantik
- Normalformen
- Unentscheidbarkeit
- Resolutionsverfahren

- Wichtige Grundbegriffe werden hier natürlich wiederholt. Aber das wird SEHR schnell gehen.
- Sehen Sie sich die Unterlagen aus dem Grundstudium noch mal an!

Lernziele dieser Vorlesung

Teil I: Erweiterungen der klassischen Logik

Mathematische Logik als Disziplin

- Fachterminologie, allgemeine Prinzipien und Verfahren
 - Leistungen, Anwendungsbedingungen, Ablauf der Verfahren
 - Uniforme Notation, Tableau-Verfahren

Logiken für die Beschreibung von Programmen

- Grundprinzipien und Eigenschaften jeweils eines Dialektes.
- Aussagen- und Prädikatenlogik: **Semantik & Beweistheorie**
 - Tableau-Kalkül
 - Exkurs zu mehrwertigen Logik
- Modallogische Aussagenlogik
 - Zeitlogik, Beschreibungslogiken als Spezialfälle
- Sortenlogik, Typenlogik, Lambda-Kalkül
 - Strukturierte Domänen, komplexe Funktions- und Relationsausdrücke

Lernziele dieser Vorlesung

Teil II: Semantik von Programmiersprachen

Denotationale Semantik, Domains, Stetigkeit, Fixpunktsatz

- Domains, Stetigkeit, Monotonie, Funktionen und Funktionale
- Fixpunktsatz (Kleene, Tarski/Knaster)
- zusammengesetzte Datenstrukturen

Axiomatische Semantik

- Hoare-Tripel, Beweiskalküle

Auswertungsstrategien funktionaler Sprachen

- eager-evaluation: alles sofort auswerten
- normal-order-evaluation: Operandenauswertung etc. erst wenn nötig
- lazy-evaluation: 2.+ keine doppelte Auswertungen von Gleichem

Logik und Semantik

Organisatorisches

Logik & Semantik – Curriculare Organisation

Gemeinsame Planung und Durchführung durch die Arbeitsbereiche WSV und TGI

Teil 1: Erweiterungen
der klassischen Logik

WSV

Christopher Habel
Carola Eschenbach

Teil 2: Semantik von
Programmiersprachen

TGI

Rüdiger Valk
Matthias Jantzen

SoSe
2007

Prüfungen

sind studienbegleitend konzipiert !

- während des Semesters lernen und fragen,
- am Semesterende prüfen,
- und nicht erst 2 Jahre später !

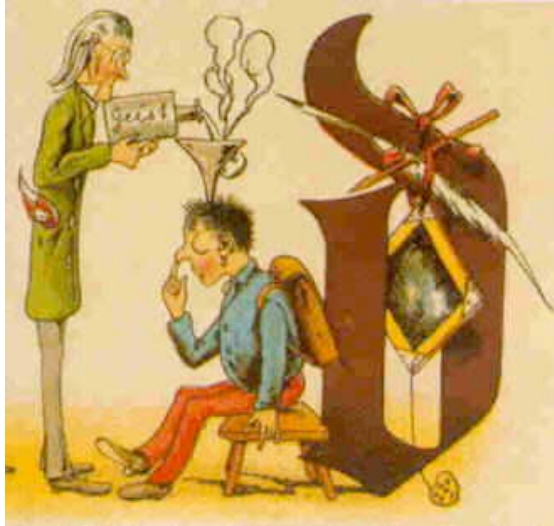
Inhalt

- Stoff der gesamten Vorlesung
- ggf. auch spezielles Logik-Buch (vorher absprechen)

Termine (über das Prüfungsamt zu vergeben)

- werden zur Semestermitte bekannt gegeben

Prüfungsvorbereitung



Ch. Habel / C. Eschenbach: Logik & Semantik

1 – 18

Prüfungsvorbereitung (auch gemeinsam !)

Überblick verschaffen

- Umfang; Was wurde behandelt (unter welcher Perspektive)
- Zusammenhänge (zwischen Logiken und zwischen den Inhalten der beiden Vorlesungsteile)

Geeignetes Material zusammenstellen

- Folienkopien allein sind keine gute Grundlage
- **eigene Mitschriften und Notizen**, Bücher, Handbucharikel

Stoff und Techniken lernen, z.B.

- Tableau-Expansion **SELBST** durchführen, Programm bei der Expansion ‚beobachten‘, Ergebnis der Expansion interpretieren / erläutern
- Einsatzbereich, Grundidee, Formate einzelner Logiksysteme
- über die Inhalte (auf Deutsch !) reden / schreiben
- Selbsteinschätzung: eigene Stärken kennen

Zum Vorgehen / Organisatorisches

„Logik & Semantik“

- Erfordert Vor-, Mit- und Nacharbeiten **während des Semesters**
- Mitarbeit in den Übungen ist essentiell für den Lern- und Prüfungserfolg.
- Sollte auf dieser Basis „studienbegleitend prüfbar“ sein.

Teil I: Vorlesung mit integrierten Übungen

- Kurzübungen (ca. 5. Min.) in der Vorlesung (**Papier & Stift!!!**)
 - Schriftliche Übungen zwischen den Vorlesungsterminen
 - Programm-Pakete (Prolog) zum Nachvollziehen, Ausprobieren, Erweitern der vorgestellten Verfahren
- Ergänzende Aufgaben in den Prüfungsunterlagen (Folienkopien)
- www.informatik.uni-hamburg.de/WSV/teaching/vorlesungen/LOS_SoSe07.php
- FBI → Gliederung → WSV → Lehre → LOS

- Elektronische Unterlagen
 - Zugang über <http://www.informatik.uni-hamburg.de/WSV/teaching/> pdf-Dateien der Folien (ggf. aktualisiert)
- Folien sind keine Skripte !
 1. Mitdenken und mitschreiben ist wichtig (Abschreiben von Folien natürlich nicht.)
 2. Lesen von Lehrbüchern und Handbuchaufsätzen ist notwendig, dabei müssen insbesondere die Beziehungen zwischen den verschiedenen Notationen und Terminologien hergestellt werden.
 3. Machen Sie die Übungsaufgaben, um Ihr Verständnis zu schärfen.

Schriftliche Übungen zwischen den Vorlesungsterminen

Wöchentlich soll ein Übungsblatt bearbeitet werden.

- Ausgabe jeweils Donnerstags (erstes Blatt am 5. April 2007)
- Abgabe jeweils am darauf folgenden Montag.

Korrektur / Durchsicht der Übungsblätter (keine Benotung!)

- Alle, die Übungsblätter abgeben, müssen ein Übungsblatt korrigieren / durchsehen. (Jeweils von Montag bis Donnerstag.)
- Diese Korrekturen / Kommentare werden noch einmal von mir (C.E.) angesehen.

Ziel der Übungen und Korrektur der Übungen:

- aktives Arbeiten mit dem Stoff der Vorlesung
- Schreiben sowie Lesen und Bewerten „logischer Texte“

Gelingen ist vom Mitwirken der Beteiligten abhängig

Programm-Pakete (Prolog)

Zum Ausprobieren (für alle Beteiligten)

- Bereitstellung auf LOS-Veranstaltungsseite
 - Zip-Dateien zum runterladen, auspacken → Ordner mit Prolog-Programmen und Erläuterungen
- SWI-Prolog
 - Läuft auf den Unix-Maschinen des FB Informatik
 - Erhältlich von <http://www.swi-prolog.org/> für verschiedene Plattformen
- Rückmeldungen erbeten !! (auch schon frühzeitig)

Angebot, keine Verpflichtung

- Wer im ersten Semester Prolog gelernt hat, kann hier einmal wieder damit arbeiten.

- Bitte um Rückmeldung: Einerseits darüber, wer welche Version ausprobiert hat. Andererseits zu Erfahrungen und Ergänzungsmöglichkeiten.
- Bitte um Ergänzungen: die Materialien können nicht perfekt sein. Insbesondere sind die Listen mit Beispielen zu kurz geraten. Wer diese Listen in interessanter Weise ergänzt, kann mir die Ergänzungen gerne zusenden, damit ich sie anderen zur Verfügung stellen kann.

Logik und Semantik

Logik

Logik: Gegenstand und Zielsetzung

Was untersucht die Logik?

- Schlüsse und Argumente
 - insbesondere ihre formale Rechtfertigung
- Beweise und ihre Struktur (oft als ‚Form‘ bezeichnet)
 - Kommunikation von Begründungszusammenhängen
- Sprachen,
 - in denen die Welt beschrieben werden kann, so dass
 - mit den Beschreibungen formal gerechtfertigt geschlossen werden kann
 - über diese Beschreibungen rational argumentiert werden kann
- Theorien und ihre Konsequenzen
 - Vorhersagekraft einer Theorie
 - Bewertung der Theorie über die Bewertung ihrer Konsequenzen

Was betrachtet die Logik nicht?

Gegenstandsbereiche,

- über die argumentiert wird (Mathematik, Ethik, Biologie)
- Unterschiede zwischen Gegenstandsbereichen
- Korrektheit von Aussagen und Theorien für einen Gegenstandsbereich

Definition eines logischen Systems: Generelles Schema

Spezifikation

- einer formalen Sprache (zur Repräsentation)
- von Evaluations- / Interpretationsprinzipien
- semantischer Kategorisierungen und Beziehungen
- Ableitungs-, Beweisverfahren

Historisch gesehen ist die Entwicklung von Beweisverfahren den anderen Schritten vorgelagert. Die Mathematiker und Philosophen haben Beweise geführt, lange bevor die formal-sprachlichen und semantischen Theoriegebäude entwickelt waren. Wahrheitstafeln sind eine relativ moderne Errungenschaft. Dies spricht dafür, dass die Menschen über ein ‚natürliches‘ Verständnis von Logik und Beweisen verfügen.

Eine Aufgabe der philosophischen Logik ist es, dieses ‚natürliche‘ Verständnis zu einem analytischen Werkzeug zu entwickeln, mit dem Argumentationsketten und Gedankengänge auf ihre Überzeugungskraft hin beurteilt werden können.

Eine Aufgabe der Mathematischen Logik ist es, die natürliche Logik formal so zu fassen, dass sie durch Maschinen verarbeitbar ist und die Maschinen die den Menschen ermüdende Aufgabe der detaillierten Beweisführung und -prüfung abnehmen können.

Logische Sprache

- ‚logische‘ Basisbausteine (logische Konstanten, Junktoren)
- Variablen
- ‚nicht-logische‘, ‚freie‘ Basisbausteine, ‚verfügbare‘ Symbole
- Hilfssymbole (Klammern, Interpunktion)
- Syntaktische Unterscheidungen
 - Formeln (alle Logiken)
 - Terme (Prädikatenlogik, Lambda-Kalkül, Beschreibungslogiken)
 - Relations-, Funktionsausdrücke (Prädikatenlogik, Beschreibungslogiken, Lambda-Kalkül)
- Induktive Definition der Gesamtmenge der Ausdrücke
 - atomare Ausdrücke
 - komplexe, zusammengesetzte Ausdrücke
 - Legitimation Induktiver Beweise und rekursiver Funktionsdefinitionen

Evaluations- / Interpretationsprinzip

- Wahrheitswerte (zwei oder mehr)
- Domänen / Universen / Individuenmenge, mögliche Welten
- Syntaktische Kategorien entsprechen Evaluationsbereichen
 - Formeln → Wahrheitswerte
 - Terme → Individuen
 - Relationsausdrücke → Mengen von (Tupeln von) Individuen
 - Funktionsausdrücke → Funktionen
- **Evaluation von Ausdrücken**
 - feste Interpretation logischer Basisbausteine
 - ‚freie Interpretation‘ nicht-logischer Basisbausteine
 - kontextabhängige Interpretation von Variablen
 - rekursive Funktion zur Interpretation komplexer Ausdrücke (orientiert an den syntaktischen Bildungsprinzipien)

Semantische Kategorien und Beziehungen

- Basierend auf den Interpretationen
 - unter Berücksichtigung **aller möglichen freien Interpretationen** nicht-logischer Basisbausteine
- Klassifikation von Formeln
 - **Tautologie, Gültigkeit, Allgemeingültigkeit**
 - **Erfüllbarkeit – Falsifizierbarkeit**
 - **Kontradiktion, Unerfüllbarkeit**
- Beziehung zwischen Formeln und Formelmengen
 - **Äquivalenz**
 - **Folgerung**
- Dabei sollte erreicht werden (zum Beispiel)
 - Die **Negation** einer **Tautologie** ist eine **Kontradiktion**.
 - Alle **Kontradiktionen** sind **äquivalent**.

Ableitungs-, Beweisverfahren

- Basierend auf der syntaktischen Struktur der Sprache
- Klassifikation von Formeln / Formelmengen
 - Beweisbarkeit, Theorem, Ableitbarkeit
 - Konsistenz
 - Inkonsistenz, widersprüchlich
- Beziehung zwischen Formeln und Formelmengen
 - Ableitbarkeit
- Dabei sollte erreicht werden:
 - **Korrektheit**
 - **Vollständigkeit**

- **Korrektheit**
 - Nur Tautologien sind beweisbar, Theorem.
 - Nur Kontradiktionen sind inkonsistent, widersprüchlich.
 - Nur folgerbare Formeln sind ableitbar.
- **Vollständigkeit**
 - Alle Tautologien sind beweisbar, Theorem.
 - Alle Kontradiktionen sind inkonsistent, widersprüchlich.
 - Alle folgerbare Formeln sind ableitbar.

Beweis

Ein Beweis ist eine korrekte, vollständige Argumentation

- weist die Wahrheit **einer Behauptung** unter **Voraussetzung der Annahmen** nach,
- dient der Kommunikation von **Begründungszusammenhängen**,
- ist ein Text,
- macht alle Annahmen explizit,
- zeigt eine Argumentationslinie (**Sequenz von Argumentationsschritten**) auf,
 - die Argumentationslinie beginnt mit den Annahmen und endet mit der Behauptung
 - jeder Schritt ist legitimiert / begründet
 - die Argumentationslinie kann Hilfsbeweise enthalten.

- Beweise, die von Menschen für Menschen gemacht sind, sehen anders aus, als formale Beweise, die von und für Maschinen gemacht sein können.
- Für die Korrektheit eines formalen Beweises gibt es formale Kriterien.
- Ein ‚menschlicher‘ Beweis gilt als korrekt, wenn er sich in einen formalen Beweis überführen lässt.

Umgebung und Struktur eines Beweises

Voraussetzungen

- Annahmen, Definitionen

Behauptung

- Aussage, deren Wahrheit zu beweisen ist

Beweis

- Sequenz von Aussagen
- Eine Aussage darf nur dann in die Sequenz aufgenommen werden, wenn die Aufnahme durch vorhergehende Aussagen oder die Voraussetzungen legitimiert / begründet ist.
- Teilbeweise, z.B. auf Grund von Fallunterscheidungen
- Letzter Schritt: Begründung der Wahrheit der Behauptung

Formale Beweise in der Logik

Ein formaler Beweis in der symbolischen Logik

- Die zugrunde liegende **Sprache** ist eine formale Sprache.
- **Annahmen** und **Behauptung** sind in der formalen Sprache ausgedrückt.
- Der **Beweis** ist eine Sequenz von Ausdrücken der formalen Sprache.
- **Legitimationen / Begründungen** für die Anfügung von Aussagen beruhen auf (festgelegten) formalen Regeln.

Ziel der Logik (als wissenschaftliche Disziplin)

- Theorie der Beweise, Argumentationen, Schlüsse, die allgemein anerkannte Beweise auch als solche ausweist

Literatur

- Melvin Fitting (1996). *First-order logic and automated theorem proving*. New York: Springer. 2nd Edition.
- Steffen Hölldobler (2001). *Logik und Logikprogrammierung*. Heidelberg: Synchron. 2. Erweiterte Auflage.
- Mordechai Ben-Ari (2001). *Mathematical Logic for Computer Science*. Springer: London. 2nd Edition.
- Nerode, Anil & Richard A. Shore (1997). *Logic for Application*. New York: Springer.
- John C. Reynolds (1998). *Theories of programming languages*. Cambridge: Cambridge University Press.
- R.D. Tennent (1994). Denotational semantics. In S. Abramsky, D.M. Gabbay & T.S.E. Maibaum (eds.) *Handbook of Logic in Computer Science* (pp. 169–322). Clarendon Press: Oxford.
- P. Pepper (1992). *Grundlagen der Informatik*. Oldenbourg Verlag.
- + weitere – kürzere – Kapitel aus anderen Büchern bzw. Aufsätze.
- Insbesondere gibt es sehr gute Handbuch-Reihen zur Logik, die in der Informatik-Bibliothek verfügbar sind.