
Vorlesung: Grundlagen der Programmierung im Sommersemester 04

Dozent: [Ralf Möller](#)

Zeit: Mi 8.00-9.15, Mi 9.30-10.45 Uhr

Ort: FH-Wedel Hörsaal 4

[Uebung \(betreut von Alissa Kaplunova\)](#)

Zeit: Mo 12.30 Uhr

Ort: Hörsaal 4

Die Vorlesung ist eine Einstiegsveranstaltung in die Informatik für das 1. Semester (insgesamt 24 Vorlesungen). Für eine Einführung in die praktische Programmierung gibt es die Vorlesung [Programmiersprachen 1](#). Beide Veranstaltungen ergänzen sich. Eventuelle Überlappung geringen Ausmaes sind notwendig, um die Veranstaltungen in sich geschlossen zu halten.

In der unteren Inhaltsbersicht sind die Prsentationen zu GdP aus dem letzten Semester zu finden. Obwohl die Vorlesung in diesem Semester zum dritten Mal gehalten wird, werden die Unterlagen noch ueberarbeitet.

Ich bemhe mich, vor den jeweiligen Vorlesungen das entsprechende Material in Form von PDF-Dokumenten bereitzustellen (s.u.).

Die Dokumente sind z.B. mit dem Programm [Acroread](#) lesbar.

Achtung: Es handelt sich bei den untenstehenden Versionen in jedem Fall um vorlufige Versionen der Dokumente.

Fehlerkorrekturen, Aenderungen und auch Ergnzungen sind jederzeit mglich.

Bitte beachten Sie, dass auch die Kopiertversion der Praesentationen beim Asta nicht die endgltige Version der Vorlesungsunterlagen darstellt.

Voraussetzungen:

Mengenlehre, Funktionen, Relationen

Inhalt:

- **Einfhrung: Algorithmen, Entwurf von Algorithmen**
([Vorlesung 1](#))
([komprimierte Version](#))
- **Aussagenlogik**
Syntax, Semantik, Modellbegriff, Erfllbarkeit, Allgemeingltigkeit, Aequivalenz, Anwendung: Angabe von Bedingungen, Transformation von Bedingungen
([Vorlesung 23](#))
- **Aussagenlogik**
Folgerbarkeit und Aequivalenz

Entscheidungsverfahren für die aussagenlogische Äquivalenz und Folgerbarkeit

([Vorlesung 33](#))

- **Ersetzungstheorem, Boolesche Algebra**
Äquivalenztransformation von aussagenlogischen Formeln,
([Vorlesung 4](#))
- **Normalformen**
Transformationsalgorithmen
([Vorlesung 5](#))
- **Resolution, Vollständigkeit und Korrektheit**
Anwendungen: Lösen von Logeleien
([Vorlesung 6](#))
- **Prädikatenlogik Teil 1**
Motivation, Syntax
([Vorlesung 7](#))
- **Prädikatenlogik Teil 2**
Semantik, Modellbegriff
Entscheidungsprobleme: Erfüllbarkeit, Allgemeingültigkeit
([Vorlesung 8](#))
- **Prädikatenlogik Teil 3**
Entscheidungsprobleme: Folgerbarkeit, Äquivalenz
Anwendung: Spezifikation von Daten
([Vorlesung 9](#))
- **Systematische Programmentwicklung Teil 1**
Prädikatenlogik mit speziellen Theorien, Variablen und Felder, Entwicklung von Spezifikationen
([Vorlesung 10](#))
- **Systematische Programmentwicklung Teil 2**
Von der Spezifikation zum Programm, Korrektheit von Programmen definiert über Vor- und Nachbedingungen
Einfach- und Mehrfachzuweisungen, schwächste Vorbedingungen
([Vorlesung 11](#))
- **Systematische Programmentwicklung Teil 3**
Fallunterscheidungen, schwächste Vorbedingungen
([Vorlesung 12](#))
- **Systematische Programmentwicklung Teil 4**
Schleifen: Invarianten, schwächste Vorbedingungen
([Vorlesung 13](#))
- **Systematische Programmentwicklung Teil 5**
Schleifen: Terminierung
([Vorlesung 14](#))
- **Systematische Programmentwicklung Teil 6**
Funktionen, Blöcke, Prozeduren, Rekursion
Auswertestrategien, Rekursionsformen
([Vorlesung 15](#))
- **Systematische Programmentwicklung Teil 7**
Umwandlung von Rekursion in Iteration
Korrektheit und Terminierung von rekursiven Programmen
([Vorlesung 16](#))
- **Analyse von Algorithmen, Ressourcen: Platz und Zeit**
Charakterisierung der Komplexität von Funktionen,
Asymptotischer Aufwand, O-Notation
([Vorlesung 17](#))
- **Analyse von Algorithmen, Zeitkomplexität**
Sortierverfahren
([Vorlesung 18](#))
- **Formale Sprachen, Grammatiken, Grammatiktypen**
([Vorlesung 19](#))
- **Formale Sprachen, Entscheidungsprobleme, Ableitungsgraphen, Eindeutigkeit**
([Vorlesung 20](#))
- **Endliche Automaten, Reguläre Sprachen**
([Vorlesung 21](#))
- **Entscheidbarkeit von Typ-1-Sprachen**
([Vorlesung 22](#))
- **Reguläre Ausdrücke, Eigenschaften Regulärer Sprachen, Kontextfreie Sprachen, CYK-Algorithmus**
Anwendungen im Compilerbau
([Vorlesung 23/24](#))

Die SoSe04-Version der Vorlesung baut auf einer früheren Version der Vorlesung von [Uwe Schmidt](#) auf.

Weiteres Material aus dem Buch "Logik für Informatiker" wurde aus der SoSe02-Version der Vorlesung [Logik](#) übernommen.

Das Material zur asymptotischen Komplexität wurde aus einer SoSe02-Version der Vorlesung

[Einführung in die Informatik 1](#) der Universität-Gesamthochschule Siegen übernommen.

Die SoSe02-Version der Vorlesung 16 über die Analyse von Sortieralgorithmen ist an das Material der SoSe02-Version der Vorlesung

[Datenstrukturen und Algorithmen](#) von der RWTH Aachen angelehnt.

Die SoSe02-Version der Vorlesungen 17 und 24 beziehen sich auf das Buch "Theoretische Informatik kurz gefaßt" von Uwe Schöning.

Die Präsentationen sind an die SoSe02-Version der Vorlesung [Einführung in die Informatik IV - Theoretische Informatik](#) angelehnt.

Klausuren

- [Uebergangsprüfung SoSe 02 \(Musterlösung\)](#)
 - [Nachprüfung Uebergangsprüfung SoSe 02](#)
 - [Uebergangsprüfung WS 02/03 \(Musterlösung\)](#)
 - [Uebergangsprüfung 2 WS 02/03](#)
 - [Klausur Sommer 03](#)
 - [Klausur WS 03/04](#)
-

Literatur:

- Vorlesungen 2-7:
Uwe Schöning, Logik für Informatiker,
Spektrum Akad. Vlg., Hdg. Erscheinungsdatum: 2000, Auflage: 5. Aufl., ISBN: 3827410053
- Vorlesungen 8-14:
M. Huth, M. Ryan, Logic in Computer Science, Modeling and Reasoning about Systems,
Cambridge University Press, 2000
- Vorlesungen 15-16:
G. Goos, Vorlesungen über Informatik Band 1, Grundlagen und funktionales Programmieren, Springer-Verlag, 3. Auflage,
2000
- Vorlesungen 17-18:
T.H. Cormen, C.E. Leiserson, R.L. Rivest, Introduction to Algorithms, MIT Press, 1990
- Vorlesung 19-24:
Uwe Schöning, Theoretische Informatik kurz gefaßt,
Spektrum Akad. Vlg., Hdg. Erscheinungsdatum: 2001, Auflage: 4. Aufl., ISBN: 3827410991

Aus den genannten Werken sind jeweils nur Teilkapitel relevant.

[Ralf Möller](#)

Last modified: 24.5.04