



**Universität Stuttgart**

**Modulhandbuch**  
**Studiengang Bachelor of Science Informatik**  
Prüfungsordnung: 2009

Sommersemester 2013  
Stand: 11. April 2013

Universität Stuttgart  
Keplerstr. 7  
70174 Stuttgart

---

## Kontaktpersonen:

---

Studiendekan/in:	Univ.-Prof.Dr. Daniel Weiskopf Institut für Visualisierung und Interaktive Systeme Tel.: E-Mail: daniel.weiskopf@vis.uni-stuttgart.de
Studiengangsmanager/in:	Corinna Vehlow Visualisierungsinstitut der Universität Stuttgart Tel.: E-Mail: corinna.vehlow@visus.uni-stuttgart.de
Prüfungsausschussvorsitzende/r:	Univ.-Prof.Dr. Otto Eggenberger Institut für Rechnergestützte Ingenieursysteme Tel.: E-Mail: otto.eggenberger@iris.uni-stuttgart.de
Fachstudienberater/in:	Bernhard Schmitz Institut für Visualisierung und Interaktive Systeme Tel.: E-Mail: Bernhard.Schmitz@vis.uni-stuttgart.de

# Inhaltsverzeichnis

<b>Präambel</b> .....	<b>4</b>
<b>Qualifikationsziele</b> .....	<b>5</b>
<b>100 Basismodule</b> .....	<b>6</b>
12060 Datenstrukturen und Algorithmen .....	7
10190 Mathematik für Informatiker und Softwaretechniker .....	9
10260 Programmierkurs .....	11
10280 Programmierung und Software-Entwicklung .....	12
10930 Technische Grundlagen der Informatik .....	14
10940 Theoretische Grundlagen der Informatik .....	16
<b>200 Kernmodule</b> .....	<b>18</b>
10020 Algorithmik .....	19
14910 Berechenbarkeit und Komplexität .....	21
25610 Grundlagen des Software Engineerings .....	23
10210 Mensch-Computer-Interaktion .....	24
10220 Modellierung .....	26
10240 Numerische und Stochastische Grundlagen .....	28
10310 Rechnerorganisation .....	30
40090 Systemkonzepte und -programmierung .....	32
<b>300 Ergänzungsmodule</b> .....	<b>34</b>
320 Katalog ISG 1-3 .....	35
330 Katalog ISW 1-3 .....	36
340 Katalog ISW 4-7 .....	37
350 Wahlmodule aus Master Informatik .....	38
<b>400 Schlüsselqualifikationen fachaffin</b> .....	<b>39</b>
10290 Projekt-INF .....	40
10320 Seminar-INF 1 .....	42
42390 Seminar-INF 2 .....	44

## Präambel

Die Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik beschreibt den Aufbau des Studiums und die Organisation der Prüfungen. Sie stellt das Regelwerk und die Rechtsgrundlage für eine einheitliche Handhabung des Studienablaufs und der Bewertung der Studien- und Prüfungsleistungen dar. Sie wendet sich dabei sowohl an die Studierenden als auch an die Prüfenden sowie an die entsprechenden Organe der Universität Stuttgart.

Aus Gründen der sprachlichen Vereinfachung wird grundsätzlich nur die grammatikalisch männliche Form für Personen verwendet. Sinngemäß ist stets auch die entsprechende weibliche Form gemeint.

## Qualifikationsziele

Das Curriculum des Studienganges sieht im 1. und 2. Semester eine Vertiefung der mathematischen und technischen Grundlagen der Informatik in Pflichtmodulen vor. Im 3. bis 6. Semester liegt der Schwerpunkt der Ausbildung auf diversen Grundlagen- und Spezialisierungsfächern verschiedener Bereiche der Informatik. Zusätzliche Inhalte sind Seminare, fachliche Module als Wahlpflichtbereich sowie fachübergreifende Schlüsselqualifikationen. Im 6. Semester liegt ein weiterer Schwerpunkt auf der Bachelorarbeit. Mit der Bachelorarbeit ist die Befähigung zu zeigen, innerhalb einer vorgegebenen Frist eine komplexe Aufgabenstellung aus dem Bereich der Informatik selbständig nach wissenschaftlichen Methoden zu bearbeiten und die Ergebnisse sachgerecht darzustellen.

Die Absolventinnen und Absolventen des Bachelorstudienganges Informatik

- verfügen über ein vertieftes mathematisches und ingenieurwissenschaftliches Wissen, welches sie befähigt, Probleme und Aufgabenstellungen der Informatik zu verstehen und kritisch einzuschätzen.
- sind in der Lage, jenes erlangte Wissen auf Erkenntnisse der Ingenieurwissenschaften anzuwenden.
- verfügen über Fachwissen auf den Gebieten der theoretischen, praktischen, technischen und angewandten Informatik und können Aufgabenstellungen der Informatik wissenschaftlich erkennen, beschreiben und bewerten, analysieren und lösen.
- haben umfassendes Verständnis über Softwareentwicklungsmethoden und Rechnersysteme, ihre Anwendungsmöglichkeiten und verfügen über die Fertigkeit, Konzepte und Lösungen für neue Softwaresysteme zu entwickeln.
- verfügen über eine verantwortliche und selbständige wissenschaftliche Arbeitsweise.

Die Beschäftigungsbereiche der Absolventinnen und Absolventen liegen u. a. in der Datenverarbeitungs- und Entwicklungsindustrie und in industriellen, kommerziellen, verwaltungstechnischen und wissenschaftlichen Anwendungen von Datenverarbeitungssystemen.

---

## 100 Basismodule

---

Zugeordnete Module:    10190 Mathematik für Informatiker und Softwaretechniker  
                              10260 Programmierkurs  
                              10280 Programmierung und Software-Entwicklung  
                              10930 Technische Grundlagen der Informatik  
                              10940 Theoretische Grundlagen der Informatik  
                              12060 Datenstrukturen und Algorithmen

---

## Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Thomas Ertl		
9. Dozenten:	Stefan Funke		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 2. Semester → Basismodule B.Sc. Informatik, PO 2012, 2. Semester → Basismodule BA (Komb) Informatik, PO 2009, 2. Semester → Module im Nebenfach		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 051520005 Programmierung und Software-Entwicklung</li> </ul>		
12. Lernziele:	<p>Die Studierenden kennen nach engagierter Mitarbeit in dieser Veranstaltung diverse zentrale Algorithmen auf geeigneten Datenstrukturen, die für eine effiziente Nutzung von Computern unverzichtbar sind. Sie können am Ende zu gängigen Problemen geeignete programmiersprachliche Lösungen angeben und diese in einer konkreten Programmiersprache formulieren.</p> <p>Konkret:</p> <ul style="list-style-type: none"> <li>• Kenntnis der Eigenschaften elementarer und häufig benötigter Algorithmen</li> <li>• Verständnis für die Auswirkungen theoretischer und tatsächlicher Komplexität</li> <li>• Erweiterung der Kompetenz im Entwurf und Verstehen von Algorithmen und der zugehörigen Datenstrukturen</li> <li>• Erste Begegnung mit nebenläufigen Algorithmen; sowohl „originär“ parallel, als auch parallelisierte Versionen bereits vorgestellter sequentieller Algorithmen</li> </ul>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen</li> <li>• Komplexität und Effizienz von Algorithmen, O-Notation</li> <li>• Wahl der Datenstrukturen; Listen, Bäume, Graphen; deren Definitionen, deren Datenstrukturen</li> <li>• diverse interne und externe Such- und Sortierverfahren (z.B. Linear-, Binär-, Interpolationssuche, AVL-, B-Bäume, internes und externes Hashing, mehrere langsame Sortierungen, Heap-, Quick-, Bucket-, Mergesort)</li> <li>• diverse Graphenalgorithmen (DFS, BFS, Besuchssequenzen, topol. Traversierung, Zusammenhangskomponenten, minimale Spannbäume, Dijkstra-, Floyd- kürzeste Wege)</li> <li>• Algorithmen auf Mengen und Relationen (transitive Hüllen, Warshall)</li> <li>• Korrektheitsbegriff und -formalismen; Spezifikation und Implementierung</li> <li>• Einige parallele und parallelisierte Algorithmen</li> <li>• einfache Elemente paralleler Programmierung, soweit für obiges notwendig</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Appelrath H.J., Ludewig. J., Skriptum Informatik, 1999</li> <li>• Sedgewick, R., Algorithms in C, 1998</li> </ul>		

---

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 120601 Vorlesung Datenstrukturen und Algorithmen</li><li>• 120602 Übung Datenstrukturen und Algorithmen</li></ul>
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 12061 Datenstrukturen und Algorithmen (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben.</li><li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li></ul>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

---

## Modul: 10190 Mathematik für Informatiker und Softwaretechniker

2. Modulkürzel:	080300100	5. Moduldauer:	2 Semester
3. Leistungspunkte:	18.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	12.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof.Dr. Wolfgang Rump		
9. Dozenten:	Wolfgang Rump		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 1. Semester → Basismodule B.Sc. Informatik, PO 2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine, die Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Studierenden haben die mathematischen Grundlagen für die Studiengänge Informatik bzw. Softwaretechnik erarbeitet und den selbständigen und kreativen Umgang mit den mathematischen Stoffgebieten gelernt.		
13. Inhalt:	1. Semester: <ul style="list-style-type: none"> <li>• Grundlagen (Aussagenlogik, Mengen, Relationen, Abbildungen, Zahlenmengen, Grundbegriffe der Algebra)</li> <li>• Lineare Algebra (Vektorräume, lineare Abbildungen, Matrizen, Determinanten, lineare Gleichungssysteme, Eigenwerte, Normalformen, Hauptachsentransformation, Skalarprodukte)</li> <li>• Analysis (Konvergenz, Zahlenfolgen und Zahlenreihen, stetige Abbildungen, Folgen und Reihen von Funktionen, spezielle Funktionen)</li> </ul> 2. Semester: <ul style="list-style-type: none"> <li>• Differential- und Integralrechnung (Funktionen einer und mehrerer Variablen, Ableitungen, Taylorentwicklungen, Extremwerte, Integration, Anwendungen)</li> <li>• Gewöhnliche Differentialgleichungen (elementar lösbare Differentialgleichungen, Existenz und Eindeutigkeit von Lösungen)</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Anna Sändig, Mathematik, Vorlesungskripte , SS 2007</li> <li>• D. Hachenberger, Mathematik für Informatiker, 2005</li> <li>• M. Brill, Mathematik für Informatiker, 2001</li> <li>• P.Hartmann, Mathematik für Informatiker, 2002</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101901 Vorlesung Mathematik 1 für Informatik und Softwaretechnik</li> <li>• 101902 Übung Mathematik 1 für Informatik und Softwaretechnik</li> <li>• 101903 Vorlesung Mathematik 2 für Informatik und Softwaretechnik</li> <li>• 101904 Übung Mathematik 2 für Informatik und Softwaretechnik</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 126 Stunden Nachbearbeitungszeit: 414 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10191 Mathematik für Informatiker und Softwaretechniker (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Ein Übungsschein aus den beiden Veranstaltungen, jeweils im 1. oder 2. Fachsemester zu erwerben</li> <li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li> </ul>		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

---

## Modul: 10260 Programmierkurs

2. Modulkürzel:	051520010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof.Dr. Stefan Wagner		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Max Kisselew</li> <li>• Ivan Bogicevic</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 1. Semester → Basismodule  BA (Komb) Informatik, PO 2009, 1. Semester → Module im Nebenfach		
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	Selbstständiges Erstellen von Programmen und Lösung von Programmieraufgaben in einer vorgegebenen Programmiersprache wie Java.		
13. Inhalt:	Der Programmierkurs richtet sich an die Studierenden im BSc. Maschinelle Sprachverarbeitung. Vermittelt wird die Programmiersprache Python.		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	102601 Übung Programmierkurs		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17. Prüfungsnummer/n und -name:	10261 Programmierkurs (USL), Sonstiges, Gewichtung: 1.0, Übungsschein - Scheinkriterien werden zu Beginn der Veranstaltung angekündigt.		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

## Modul: 10280 Programmierung und Software-Entwicklung

2. Modulkürzel:	051520005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof.Dr. Stefan Wagner		
9. Dozenten:	Bernhard Mitschang		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 1. Semester → Basismodule B.Sc. Informatik, PO 2012, 1. Semester → Basismodule BA (Komb) Informatik, PO 2009, 1. Semester → Module im Nebenfach		
11. Empfohlene Voraussetzungen:	Keine. Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Teilnehmer haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Die Programmiersprache Java und die virtuelle Maschine</li> <li>• Objekte, Klassen, Schnittstellen, Blöcke, Programmstrukturen, Kontrakte</li> <li>• Klassenmodellierung mit der UML</li> <li>• Objekterzeugung und -ausführung</li> <li>• Boolesche Logik</li> <li>• Verzweigungen, Schleifen, Routinen, Abstraktionen, Modularisierung, Variablen, Zuweisungen</li> <li>• Rechner, Hardware</li> <li>• Syntaxdarstellungen</li> <li>• Übersicht über Programmiersprachen und -werkzeuge</li> <li>• Grundlegende Datenstrukturen und Algorithmen</li> <li>• Vererbung, Polymorphe</li> <li>• Semantik</li> <li>• Programmierung graphischer Oberflächen</li> <li>• Übergang zum Software Engineering</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Appelrath, Hans-Jürgen und Ludewig, Jochen, "Skriptum Informatik - eine konventionelle Einführung", Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999</li> <li>• Meyer, Bertrand, "Touch of Class", Springer-Verlag, 2009</li> <li>• Savitch, Walter, "Java. An Introduction to Problem Solving and Programming", Pearson, 6. Auflage, 2012</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 102801 Vorlesung Programmierung und Softwareentwicklung</li> <li>• 102802 Übung Programmierung und Softwareentwicklung</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	63 Stunden	

---

Vor-/Nachbearbeitungszeit: 187 Stunden

Prüfungsvorbereitung: 20 Stunden

---

17. Prüfungsnummer/n und -name:
- 10281 Programmierung und Software-Entwicklung (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Studienleistung: Übungsschein. Voraussetzungen werden zu Beginn vom Dozenten festgesetzt. Dazu gehören eine bestimmte Anzahl von Vorträgen in den Übungen und ein bestimmter Teil der Übungspunkte. Modulprüfung: schriftlich, 120 Minuten, keine Hilfsmittel
  - V Vorleistung (USL-V), schriftlich, eventuell mündlich
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

---

## Modul: 10930 Technische Grundlagen der Informatik

2. Modulkürzel:	051711005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr.-Ing. Martin Radetzki		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Martin Radetzki</li> <li>• Sven Simon</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 1. Semester → Basismodule  B.Sc. Informatik, PO 2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Elektrotechnische Grundlagen:  Der Studierende hat grundlegendes Verständnis elektrischer Schaltkreise, der Funktionsweise der Bauelemente und Komponenten von Computer-Systemen, wie Transistoren, Halbleiterschaltungen, RAM, ROM, Festplatte etc. erworben.</li> <li>• Digitaltechnische Komponenten:  Der Studierende kann digitale Schaltungen von begrenzter Komplexität analysieren, konstruieren und optimieren.</li> </ul>		
13. Inhalt:	Elektrotechnische Grundlagen: <ul style="list-style-type: none"> <li>• Elektrische Grundgrößen, Ohmsches Gesetz, Kirchhoffsche Gesetze.</li> <li>• Bauelemente: Widerstand, Kondensator, Spule, Bauelemente, Halbleiter-Leitungsmechanismen.</li> <li>• CMOS-Transistoren.</li> <li>• Integrationstechniken der Mikroelektronik.</li> <li>• Digitale Grundschaltungen, Logik- und Speicherschaltungen.</li> <li>• Technologie und Schaltungstechnik</li> <li>• Mikroprozessoren, Mikrocontroller, Signalprozessoren, FPGA.</li> </ul> Digitaltechnische Komponenten: <ul style="list-style-type: none"> <li>• Schaltalgebra, Schaltnetze / kombinatorische Netzwerke,</li> <li>• Modelle sequentiellen Verhaltens,</li> <li>• Schaltwerke / sequentielle Netzwerke,</li> <li>• Verzögerungsanalyse,</li> <li>• Taktschemata,</li> <li>• Binäre Codierung,</li> <li>• Datenpfadelemente,</li> <li>• Entwurfsmethodik und Entwurfsautomatisierung</li> </ul>		
14. Literatur:	-		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 109301 Vorlesung Elektrotechnische Grundlagen</li> <li>• 109302 Übung Elektrotechnische Grundlagen</li> <li>• 109303 Vorlesung Digitaltechnische Komponenten</li> <li>• 109304 Übung Digitaltechnische Komponenten</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden		

---

Nachbearbeitungszeit: 117 Stunden

---

17. Prüfungsnummer/n und -name:

- 10931 Technische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: In dem Fach Elektrotechnische Grundlagen werden ein oder zwei Klausuren Semester-begleitend durchgeführt. Es ist eine Mindestzahl von Punkten aus dieser Prüfung bzw. diesen Prüfungen erforderlich, um zur Prüfung Technische Grundlagen der Informatik zugelassen zu werden. Bezüglich der Elektrotechnische Grundlagen und den Digitaltechnische Komponenten ist eine Teilnahme an einer Mindestzahl der Übungen, die zu Beginn der Lehrveranstaltung festgelegt wird, erforderlich.
- V Vorleistung (USL-V), schriftlich, eventuell mündlich

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

---

## Modul: 10940 Theoretische Grundlagen der Informatik

2. Modulkürzel:	050420005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Volker Diekert</li> <li>• Ulrich Hertrampf</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 1. Semester → Basismodule B.Sc. Informatik, PO 2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Logik und Diskrete Strukturen: Die Studierenden haben die grundsätzlichen Kenntnisse in Logik und Diskreter Mathematik erworben, wie sie in den weiteren Grundvorlesungen der Informatik in verschiedenen Bereichen benötigt werden.</li> <li>• Automaten und Formale Sprachen: Die Studierenden beherrschen wichtige theoretische Grundlagen der Informatik, insbesondere die Theorie und Algorithmik endlicher Automaten. Hierzu gehört das Kennenlernen, Einordnung und Trennung der Chomskyschen Sprachklassen.</li> </ul>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Logik und Diskrete Strukturen: Einführung in die Aussagenlogik; formale Sprache; Semantik (Wahrheitswerte); Syntax (Axiome und Schlussregeln); Normalformen; Hornformeln; aussagenlogische Resolution; Korrektheit und Vollständigkeit für die Aussagenlogik; Einführung in die Prädikatenlogik 1. Stufe; formale Sprache; Semantik und Syntax; Normalformen; Herbrand-Theorie; prädikatenlogische Resolution; Kombinatorik, Graphen, elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, RSA-Verfahren.</li> <li>• Automaten und Formale Sprachen: Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, Iterationslemmata für reguläre und kontextfreie Sprachen, Normalformen, Kellerautomaten, Lösen des Wortproblems kontextfreier Sprachen mit dem CYK-Algorithmus, linear beschränkte Automaten, kontextsensitive Grammatiken, Typ 0-Grammatiken und Turingmaschinen.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988</li> <li>• Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 109401 Vorlesung Logik und Diskrete Strukturen</li> <li>• 109402 Übung Logik und Diskrete Strukturen</li> <li>• 109403 Vorlesung Automaten und Formale Sprachen</li> </ul>		

---

	• 109404 Übung Automaten und Formale Sprachen
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 84 Stunden Nachbearbeitungszeit: 276 Stunden
17. Prüfungsnummer/n und -name:	• 10941 Theoretische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Formale Methoden der Informatik

---

---

## 200 Kernmodule

---

Zugeordnete Module:

- 10020 Algorithmmik
- 10210 Mensch-Computer-Interaktion
- 10220 Modellierung
- 10240 Numerische und Stochastische Grundlagen
- 10310 Rechnerorganisation
- 14910 Berechenbarkeit und Komplexität
- 25610 Grundlagen des Software Engineerings
- 40090 Systemkonzepte und -programmierung

---

## Modul: 10020 Algorithmik

2. Modulkürzel:	050420015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Volker Diekert</li> <li>• Stefan Funke</li> <li>• Ulrich Hertrampf</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Kernmodule  B.Sc. Informatik, PO 2012, 5. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	Grundvorlesungen in theoretischer und praktischer Informatik.		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Kennenlernen und beherrschen wichtiger Programmierparadigmen und Entwurfsstrategien;</li> <li>• Selbstständiges Erarbeiten von Laufzeitabschätzungen.</li> </ul>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Entwurfsstrategien für Algorithmen (Teile und Beherrsche, Gierige Methode, Dynamische Programmierung, Backtracking, heuristische Algorithmen)</li> <li>• Analyse und Komplexität von Algorithmen</li> <li>• Mustererkennung</li> <li>• Sortierverfahren und ihre Komplexität</li> <li>• Verwaltung von Mengen</li> <li>• Union-Find-Algorithmen</li> <li>• Konvexe Hülle</li> <li>• optimale (Teil-) Bäume</li> <li>• Minimale Schnitte</li> <li>• Randomisierte Algorithmen und weitere Themen.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Alfred V. Aho, John E. Hopcroft, Jeffrey of Computer Algorithms, 1974</li> <li>• Alfred V. Aho, John E. Hopcroft, Jeffrey Algorithms, 1987</li> <li>• T. Ottmann und P. Widmayer, Algorithmen 2004</li> <li>• Thomas H. Cormen, Charles E. Leiserson, Introduction to Algorithms (Second Edition),</li> <li>• Volker Diekert, Entwurf und Analyse effizienter (Vorlesungsskript), 2006</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 100201 Vorlesung Algorithmik</li> <li>• 100202 Übung Algorithmik</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10021 Algorithmik (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvoraussetzung: Übungsschein</li> <li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li> </ul>		
18. Grundlage für ... :			

19. Medienform:

---

20. Angeboten von: Institut für Formale Methoden der Informatik

---

## Modul: 14910 Berechenbarkeit und Komplexität

2. Modulkürzel:	050420010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Stefan Funke</li> <li>• Volker Diekert</li> <li>• Ulrich Hertrampf</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 3. Semester → Kernmodule  B.Sc. Informatik, PO 2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	Inhaltliche Voraussetzungen: Theoretische Grundlagen der Informatik, Mathematik für Informatiker 1 und 2 (abgedeckt durch Pflichtmodule im Grundstudium).		
12. Lernziele:	Die Teilnehmer beherrschen wichtige theoretische Grundlagen der Informatik, können Probleme in Kategorien einordnen wie entscheidbar/unentscheidbar, effizient lösbar, deterministische/nichtdeterministische Berechnungen.		
13. Inhalt:	Gleichwertigkeit der verschiedenen Konkretisierungen des Algorithmenbegriffs, Churchsche These, Grenzen zwischen Entscheidbarkeit und Unentscheidbarkeit. Turing-Berechenbarkeit, primitiv-rekursive Funktionen, mu-rekursive Funktionen, Halteproblem, Satz von Rice, Gödelscher Satz. Wichtige Komplexitätsklassen, P-NP-Problem, NP-Vollständigkeit, Satz von Cook.		
14. Literatur:	<ul style="list-style-type: none"> <li>• Christos H. Papadimitriou, Computational Complexity , 1994</li> <li>• John E. Hopcroft, Jeffrey D. Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988</li> <li>• Volker Diekert, Komplexitätstheorie (Vorlesungsskript), 2007</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 149101 Vorlesung Berechenbarkeit und Komplexität</li> <li>• 149102 Übung Berechenbarkeit und Komplexität</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42h	
	Nachbearbeitungszeit:	118h	
	Prüfungsvorbereitung:	20h	
	<b>Gesamt:</b>	<b>180h</b>	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 14911 Berechenbarkeit und Komplexität (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein</li> <li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li> </ul>		
18. Grundlage für ... :	10020 Algorithmen		
19. Medienform:			

20. Angeboten von:

Institut für Formale Methoden der Informatik

---

## Modul: 25610 Grundlagen des Software Engineerings

2. Modulkürzel:	51520170	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof.Dr. Stefan Wagner		
9. Dozenten:	Jochen Ludewig		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 4. Semester → Kernmodule B.Sc. Informatik, PO 2012, 5. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	051520005 Programmierung und Software-Entwicklung 051510005 Datenstrukturen und Algorithmen sowie entsprechende Programmiererfahrung		
12. Lernziele:	Die Teilnehmer kennen die Grundbegriffe des Software Engineerings und haben einen Überblick über die Methoden und Techniken, die dort angewandt werden. Einige ausgewählte Methoden und Techniken können angewandt werden.		
13. Inhalt:	<p>Software Engineering kann in einer Vorlesung nicht erschöpfend behandelt werden. GSE gibt einen Überblick über das Gebiet und vertieft einzelne Themen, damit diese in der Praxis verwendet werden können. Es bildet damit auch die Basis für weitere Vertiefungen in diesem Gebiet. Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung. Die einzelnen Themen sind:</p> <ul style="list-style-type: none"> <li>• Geschichte und Konzepte des Software Engineerings</li> <li>• Der Software-Lebenszyklus und Software-Management</li> <li>• Software-Prüfung und Qualitätssicherung</li> <li>• Methoden, Sprachen und Werkzeuge für die einzelnen Phasen: Spezifikation, Grobentwurf, Feinentwurf, Implementierung, Test</li> </ul> <p>Dieses Modul kommt, wenn die Voraussetzungen erfüllt sind, auch für andere Fachrichtungen in Frage.</p>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Ludewig, Lichter: Software Engineering. dpunkt-Verlag, Heidelberg. 2. Aufl. 2010,</li> <li>• Pfleeger, Atlee: Software Engineering, Pearson. 2010</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 256101 Vorlesung Grundlagen des Software Engineerings</li> <li>• 256102 Übung Grundlagen des Software Engineerings</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138		
17. Prüfungsnummer/n und -name:	25611 Grundlagen des Software Engineerings (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:	<ul style="list-style-type: none"> <li>• Folien am Beamer unterstützt durch Tafel und Overhead</li> <li>• Dokumente, Links und Diskussionsforen in ILIAS</li> </ul>		
20. Angeboten von:	Institut für Softwaretechnologie		

## Modul: 10210 Mensch-Computer-Interaktion

2. Modulkürzel:	051900001	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof.Dr. Albrecht Schmidt		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Albrecht Schmidt</li> <li>• Thomas Ertl</li> <li>• Daniel Weiskopf</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Kernmodule</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach → Katalog ISG</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach → Katalog ISW</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• 051520005 Programmierung und Software-Entwicklung</li> <li>• 051200005 Systemkonzepte und -programmierung</li> </ul>		
12. Lernziele:	<p>Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.</p>		
13. Inhalt:	<p>Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet.</p> <p>Die folgenden Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> <li>• Einführung in die Grundlagen der Mensch-Computer Interaktion, historische Entwicklung</li> <li>• Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme</li> <li>• Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers</li> <li>• Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides</li> <li>• Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme</li> <li>• Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen</li> <li>• Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge</li> <li>• Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten</li> <li>• Akzeptanz, Evaluationsmethoden und Qualitätssicherung</li> </ul>		

---

14. Literatur:	<ul style="list-style-type: none"><li>• Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin; 2. Auflage. 2010</li><li>• Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004</li><li>• Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102101 Vorlesung Mensch-Computer-Interaktion</li><li>• 102102 Übung Mensch-Computer-Interaktion</li></ul>
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10211 Mensch-Computer-Interaktion (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein</li><li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li></ul>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

---

## Modul: 10220 Modellierung

2. Modulkürzel:	052010001	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Bernhard Mitschang</li> <li>• Frank Leymann</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Kernmodule</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach → Katalog ISG</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach → Katalog ISW</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• 051520005 Programmierung und Software-Entwicklung</li> <li>• 051510005 Datenstrukturen und Algorithmen</li> <li>• 051200005 Systemkonzepte und -programmierung</li> </ul>		
12. Lernziele:	<p>Am Ende des Moduls sind die Studierenden in der Lage, wesentliche Artefakte eines IT Systems zu modellieren. Der Zusammenhang und das Zusammenspiel solcher Artefakte ist verstanden. Die Rolle von Metamodellen und deren Erstellung ist klar.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Entity-Relationship Modell &amp; komplexe Objekte</li> <li>• Relationenmodell &amp; Relationenalgebra , Überblick SQL</li> <li>• Transformationen von ER nach Relationen, Normalisierung</li> <li>• XML, DTD, XML-Schema, Info-Set, Namensräume</li> <li>• Metamodelle &amp; Repository</li> <li>• RDF, RDF-S &amp; Ontologien</li> <li>• UML</li> <li>• Petri Netze, Workflownetze</li> <li>• BPMN</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002</li> <li>• R. Eckstein, S. Eckstein, "XML und Datenmodellierung", dpunkt.verlag 2004</li> <li>• M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work - Objektorientierte Modellierung mit UML2, 2005</li> <li>• P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure, Semantic Web, 2008</li> <li>• T.J. Teorey, Database Modeling &amp; Design, 2nd Edition, 1994</li> <li>• H.J. Habermann, F. Leymann, "Repository", Oldenbourg 1993</li> <li>• W. Reisig, "Petri-Netze", Vieweg &amp; Teubner 2010</li> <li>• B. Silver, "BPMN Method &amp; Style", Cody-Cassidy Press 2009</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 102201 Vorlesung Modellierung</li> <li>• 102202 Übung Modellierung</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden		

---

Nachbearbeitungszeit: 138 Stunden

- 
17. Prüfungsnummer/n und -name:
- 10221 Modellierung (PL), schriftlich, eventuell mündlich, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein
  - V Vorleistung (USL-V), schriftlich, eventuell mündlich
- 
18. Grundlage für ... :
- 10030 Architektur von Anwendungssystemen
  - 10080 Database and Information Systems
- 
19. Medienform:
- 
20. Angeboten von:
-

## Modul: 10240 Numerische und Stochastische Grundlagen

2. Modulkürzel:	051240005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dr. Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Dirk Pflüger</li> <li>• Stefan Zimmer</li> <li>• Marc Alexander Schweitzer</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 3. Semester          → Ergänzungsmodule          → Wahlmodule aus Master Informatik</p> <p>B.Sc. Informatik, PO 2009, 3. Semester          → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 3. Semester          → Ergänzungsmodule          → Wahlmodule aus Master Informatik</p> <p>B.Sc. Informatik, PO 2012, 3. Semester          → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 3. Semester          → Vorgezogene Master-Module</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 080300100 Mathematik für Informatiker und Softwaretechniker</li> </ul>		
12. Lernziele:	<p>Beherrschung grundlegender Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der erlernten Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.</p>		
13. Inhalt:	<p>Methoden der angewandten Mathematik, insbesondere der Numerik, Stochastik und Statistik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse:</p> <ul style="list-style-type: none"> <li>• numerische Algorithmik</li> <li>• Gleitpunktzahlen und Gleitpunkarithmetik</li> <li>• Interpolation &amp; Approximation</li> <li>• Integration</li> <li>• lineare Gleichungssysteme</li> <li>• Iterative Lösung linearer und nichtlinearer Gleichungen</li> <li>• gewöhnliche Differentialgleichungen</li> <li>• Stochastik</li> <li>• Zufall und Unsicherheit</li> <li>• diskrete und kontinuierliche Wahrscheinlichkeitsräume</li> <li>• Asymptotik</li> <li>• Elementare induktive Statistik</li> </ul> <p>Dabei wird ein konstruktiv-algorithmischer Zugang gewählt, der sich an konkreten Aufgabenstellungen aus der Informatik orientiert.</p>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Huckle, Schneider; Numerik für Informatiker</li> <li>• Schickinger T., Steger A.; Diskrete Strukturen, Band 2, 2002</li> <li>• Dahmen, Reusken; Numerik für Ingenieure</li> </ul>		

---

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102401 Vorlesung Numerische und Stochastische Grundlagen der Informatik</li><li>• 102402 Übung Numerische und Stochastische Grundlagen der Informatik</li></ul>
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10241 Numerische und Stochastische Grundlagen (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein</li><li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li></ul>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

---

## Modul: 10310 Rechnerorganisation

2. Modulkürzel:	051700005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	9.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Hans-Joachim Wunderlich		
9. Dozenten:	Hans-Joachim Wunderlich		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Für Rechnerorganisation 1: Technische Grundlagen der Informatik (10930)</li> <li>• Für Rechnerorganisation 2: Erfolgreiche Teilnahme an den Übungen von Rechnerorganisation 1 oder eine bestandene Eingangsklausur</li> </ul>		
12. Lernziele:	<p>Rechnerorganisation 1:</p> <ul style="list-style-type: none"> <li>• Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung</li> <li>• Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen</li> </ul> <p>Rechnerorganisation 2:</p> <ul style="list-style-type: none"> <li>• Kenntnisse in den Grundlagen des Entwurfs digitaler Schaltungen und Systeme,</li> <li>• Fertigkeiten in der Verwendung von Hardware-Beschreibungssprachen</li> <li>• Fertigkeiten im Umgang mit programmierbarer Logik (FPGA) und Prototypenboards,</li> <li>• Fertigkeiten im Umgang mit Werkzeugen zur Entwurfsautomatisierung,</li> <li>• Kenntnisse des Zusammenhangs von Hard- und Software</li> <li>• Erfahrung in Projektarbeit im Team</li> </ul>		
13. Inhalt:	<p>Rechnerorganisation 1:</p> <ul style="list-style-type: none"> <li>• Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Experimente mit Prozessorsimulatoren vertieft.</li> </ul> <p>Im einzelnen werden behandelt:</p> <ul style="list-style-type: none"> <li>• Grundstrukturen: Stack-, Akkumulator- und Register-basierende Rechner</li> <li>• Informationsdarstellung in Hardware: Daten- und Befehlsformate, Fehlererkennung und -korrektur, Befehlssätze und Unterstützung von Hochsprachen.</li> <li>• MIPS als RISC-Bespiel und seine Assemblerprogrammierung</li> <li>• Grundelemente und Entwurf auf Register-Transfer-Ebene</li> <li>• Grundzüge einer Hardware-Beschreibungssprache (VHDL)</li> <li>• Operationswerke: Multiplikation, Division, Gleitkommaeinheiten</li> <li>• Steuerwerksentwurf und Mikroprogrammierung</li> <li>• Befehlszyklus und Unterbrechungen</li> <li>• Pipelining und statisches Scheduling</li> </ul>		

- Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung, Segmentierung, TLB, MMU und DMA
- Leistungsbewertung: Maßzahlen und CPI, Benchmarking und einfache Anwendung von Warteschlangen

## Rechnerorganisation 2:

- Elementare Messtechnik
- Aufbau wesentlicher Bauelemente und Grundsaltungen der Elektronik.
- Entwurf eines einfachen RISC-Prozessors mit kommerziellen Entwurfssystemen.
- Umsetzung in Hardware mit einem Field Programmable Gate Array (FPGA) und einem Prototypenboard.
- Entwurf kombinatorischer und sequentieller Schaltungen.
- Arbeitstechniken zur Komplexitätsbewältigung und Konzepte zur Schaltungsvalidierung.
- Programmierung des selbst entworfenen Prozessors in Maschinensprache.

14. Literatur:	Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 103101 Vorlesung Rechnerorganisation 1</li> <li>• 103102 Übung Rechnerorganisation 1</li> <li>• 103103 Vorlesung Rechnerorganisation 2</li> <li>• 103104 Hardwarepraktikum Rechnerorganisation 2</li> </ul>
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 95 Stunden Nachbearbeitungszeit: 265 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10311 Rechnerorganisation 1 (PL), schriftliche Prüfung, 90 Min., Gewichtung: 7.0</li> <li>• 10312 Rechnerorganisation 1 - Praktikum und Übungsaufgaben (LBP), schriftlich, eventuell mündlich, Gewichtung: 3.0</li> <li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li> </ul>
18. Grundlage für ... :	<ul style="list-style-type: none"> <li>• 10090 Grundlagen der Eingebetteten Systeme</li> <li>• 10140 Advanced Processor Architecture</li> <li>• 11900 Design and Test of Systems-on-a-Chip</li> <li>• 14380 Hardware Verification and Quality Assessment</li> <li>• 29560 Algorithms and Methods in Design Automation for Micro- and Nanoelectronic Systems</li> <li>• 29610 Hardware Based Fault Tolerance</li> </ul>
19. Medienform:	
20. Angeboten von:	

## Modul: 40090 Systemkonzepte und -programmierung

2. Modulkürzel:	051200005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Kurt Rothermel</li> <li>• Frank Leymann</li> </ul>		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009 → Kernmodule</p> <p>B.Sc. Informatik, PO 2012 → Kernmodule</p> <p>BA (Komb) Informatik, PO 2009 → Module im Nebenfach → Katalog ISG</p> <p>BA (Komb) Informatik, PO 2009 → Module im Nebenfach → Katalog ISW</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>* Modul 051520005 Programmierung und Software-Entwicklung</li> <li>* Modul 051510005 Datenstrukturen und Algorithmen</li> </ul>		
12. Lernziele:	<ul style="list-style-type: none"> <li>* Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen</li> <li>* Verstehen systemnaher Konzepte und Mechanismen</li> <li>* Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.</li> <li>* Kann systemnahe Software entwerfen und implementieren.</li> <li>* Kann nebenläufige Programme entwickeln</li> <li>* Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.</li> </ul>		
13. Inhalt:	<p>Grundlegende Systemstrukturen - und organisationen</p> <ul style="list-style-type: none"> <li>• Multitaskingsystem</li> <li>• Multiprozessorsystem</li> <li>• Verteiltes System</li> </ul> <p>Modellierung und Analyse nebenläufiger Programme</p> <ul style="list-style-type: none"> <li>• Abstraktionen: Atomare Befehle, Prozesse, nebenläufiges Programm</li> <li>• Korrektheit- und Leitungskriterien</li> </ul> <p>Betriebssystemkonzepte</p> <ul style="list-style-type: none"> <li>• Organisation von Betriebssystemen</li> <li>• Prozesse und Threads</li> <li>• Eingabe/Ausgabe</li> <li>• Scheduling</li> </ul> <p>Konzepte zur Synchronisation über gemeinsamen Speicher</p> <ul style="list-style-type: none"> <li>• Synchronisationsprobleme und -lösungen</li> <li>• Synchronisationswerkzeuge: Semaphor, Monitor</li> </ul> <p>Konzepte zur Kommunikation und Synchronisation mittels Nachrichtentransfer</p> <ul style="list-style-type: none"> <li>• Taxonomie: Kommunikation und Synchronisation</li> <li>• Nachrichten als Kommunikationskonzept</li> </ul>		

- Höhere Kommunikationskonzepte

Basialgorithmen für Verteilte Systeme

- Erkennung globaler Eigenschaften
- Schnappschussproblem
- Konsistenter globaler Zustand
- Verteilte Terminierung

Praktische nebenläufige Programmierung in Java

- Threads und Synchronisation
- Socketschnittstelle
- RMI Programmierung

14. Literatur:	Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 400901 Vorlesung Systemkonzepte und -programmierung</li> <li>• 400902 Übung Systemkonzepte und -programmierung</li> </ul>
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 40091 Systemkonzepte und -programmierung (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0</li> <li>• V Vorleistung (USL-V), schriftlich, eventuell mündlich</li> </ul>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

---

## 300 Ergänzungsmodule

---

Zugeordnete Module:	320	Katalog ISG 1-3
	330	Katalog ISW 1-3
	340	Katalog ISW 4-7
	350	Wahlmodule aus Master Informatik

---

---

## 320 Katalog ISG 1-3

---

---

---

## 330 Katalog ISW 1-3

---

---

---

## 340 Katalog ISW 4-7

---

---

---

## 350 Wahlmodule aus Master Informatik

---

---

---

## 400 Schlüsselqualifikationen fachaffin

---

Zugeordnete Module:   10290 Projekt-INF  
                              10320 Seminar-INF 1  
                              42390 Seminar-INF 2

---

## Modul: 10290 Projekt-INF

2. Modulkürzel:	051900095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes Semester
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Prof.Dr. Thomas Ertl		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Schlüsselqualifikationen fachaffin B.Sc. Informatik, PO 2012, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Basismodule der Informatik. Darüber hinaus variabel je nach Projektanforderung.</li> </ul>		
12. Lernziele:	<p>Ziel dieses Moduls ist, die Studierenden frühzeitig und beispielhaft an Informatik-Forschung heranzuführen („undergraduate research“). Dazu soll in einem Team von mindestens 3 Studierenden in einem Zeitraum von höchstens 6 Monaten ein Projekt bearbeitet werden, das sich an aktuellen Forschungsfragestellungen der Abteilungen und Institute orientiert. Ein Beitrag zu laufenden Drittmittelprojekten ist möglich, ebenso eine Fortsetzung des Projekts in ausgewählten Bachelor-Thesis-Arbeiten. Die Teilnehmer können ein forschungsorientiertes Projekt unter Anleitung planen, durchführen und die Ergebnisse dokumentieren und präsentieren.</p> <p>Sie verfügen insbesondere über die folgenden generischen Kompetenzen (Schlüsselqualifikationen): Sie können in Teams an einem gemeinsamen Vorhaben arbeiten und ihre Beiträge den übergeordneten Erfordernissen anpassen. Sie sind in der Lage, ihre Ergebnisse den Projektteilnehmern vorzustellen und zu diskutieren und sie dabei gegebenenfalls auch fachfremden Teilnehmern zu erläutern. Sie können moderne Präsentations- und Visualisierungstechniken erfolgreich einsetzen.</p>		
13. Inhalt:	<p>Variabel: Es werden Projekte zu aktuellen Forschungsfragestellungen von den Prüfern des Fachbereichs Informatik angeboten. Die Themen haben einen überwiegenden Forschungscharakter, was sich aus dem Publikationspotential der erwarteten Ergebnisse ergibt. Die Projekte umfassen in der Regel: Einarbeitung und Literatursuche, Methodenentwicklung, Implementierung, Analyse, Dokumentation und Präsentation der Ergebnisse.</p> <p>Um dem Forschungscharakter des Projekts gerecht zu werden, soll das Ergebnis in einer wissenschaftlichen Publikation (max. 10 Seiten in Englisch) festgehalten werden, die einer einheitlichen Formatvorlage folgt. Einmal pro Semester sollen die bis zu einem Stichtag abgegebenen Projektpapiere auf einer internen Konferenz in einem Kurzbeitrag von den Studierenden präsentiert werden.</p>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Literatur, die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben.</li> </ul>		
15. Lehrveranstaltungen und -formen:	102901 Seminar Projekt		

16. Abschätzung Arbeitsaufwand:	180 Stunden pro Teammitglied
17. Prüfungsnummer/n und -name:	10291 Projekt-INF (USL), Sonstiges, Gewichtung: 1.0, Scheinkriterien: Aktive Mitwirkung im Projektteam. Abgabe eines Projektberichts in Form einer wissenschaftlichen Publikation (max. 10 Seiten in Englisch) gemäß einer einheitlichen Formatvorlage. Teilnahme und Mitwirkung an der internen Semesterkonferenz.
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

## Modul: 10320 Seminar-INF 1

2. Modulkürzel:	050420095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof.Dr. Daniel Weiskopf		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	Basismodule der Informatik, darüber hinaus variabel: Je nach dem gewählten Seminarthema können Vorkenntnisse aus weiteren Vorlesungen benötigt werden.		
12. Lernziele:	Die Studierenden können sich mit wissenschaftlicher Originalliteratur auseinandersetzen, deren Kernaussagen rezipieren und sich ein spezielles Thema überwiegend im Selbststudium erarbeiten. Sie sind fähig relevante Daten zu sammeln und zu interpretieren und ihre Erkenntnisse einem Fach- und Laienpublikum verständlich zu präsentieren und auf Fragen aus dem Publikum angemessen und sachgerecht zu reagieren. Sie haben gelernt, sich mit einem wissenschaftlichen Thema über einen längeren Zeitraum hinweg auseinander zu setzen und eigenständig aktuelle Hintergrundinformation zu beschaffen. Sie haben generische Kompetenzen erworben, etwa aktiv an einer wissenschaftlichen Diskussion zu einem vorher bekannten Thema teilzunehmen und durch Fragen an den Vortragenden ihr Verständnis zu erweitern. Sie können eine Diskussion leiten und moderieren und sind befähigt, ihre Ergebnisse den Seminarteilnehmern vorzustellen und mit Hilfe moderner Präsentationstechniken zu visualisieren. Sie sind in der Lage, das von ihnen erarbeitete Thema auch schriftlich darzustellen.		
13. Inhalt:	Variabel: Es werden Seminare zu diversen häufig aktuellen Themen angeboten. Das Seminar INF kann in der Informatik oder in einem affinen Fach durchgeführt werden, wie etwa Computerlinguistik, Elektrotechnik, Mathematik oder Wirtschaftswissenschaften. Welche Seminare zugelassen sind, entscheidet die Studienkommission. Zugelassene Seminare werden typischer Weise durch Aushang bekannt gegeben. Die Seminare sind in Größe und Inhalt so gestaltet, dass die generischen Kompetenzen (Schlüsselqualifikationen) der Studierenden entwickelt werden.		
14. Literatur:	Die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben.		
15. Lehrveranstaltungen und -formen:	103201 Seminar		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden <b>Gesamt: 90 Stunden</b>		
17. Prüfungsnummer/n und -name:	10321 Seminar-INF 1 (PL), Sonstiges, Gewichtung: 1.0, Scheinkriterien sind in der Regel ein Vortrag, eine schriftliche Ausarbeitung, sowie die aktive Mitarbeit während der Seminarveranstaltung.		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

---

---

**Modul: 42390 Seminar-INF 2**

---

2. Modulkürzel:	050420095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	2.0	7. Sprache:	Deutsch

---

8. Modulverantwortlicher:

---

9. Dozenten:

---

10. Zuordnung zum Curriculum in diesem Studiengang:

B.Sc. Informatik, PO 2009  
→ Schlüsselqualifikationen fachaffin

---

11. Empfohlene Voraussetzungen:

---

12. Lernziele:

---

13. Inhalt:

---

14. Literatur:

---

15. Lehrveranstaltungen und -formen:

423901 Seminar

---

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:

42391 Seminar-INF 2 (PL), Sonstiges, Gewichtung: 1.0

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

---