



# Modulhandbuch Bachelor of Science Softwaretechnik

## Inhaltsverzeichnis

<b>100</b>	<b>Basismodule</b>	<b>3</b>
10190	Mathematik für Informatiker und Softwaretechniker	4
10260	Programmierkurs	6
10280	Programmierung und Software-Entwicklung	8
10940	Theoretische Grundlagen der Informatik	11
12060	Datenstrukturen und Algorithmen	13
16520	Software-Qualität	15
17210	Einführung in die Softwaretechnik	17
<b>200</b>	<b>Kernmodule</b>	<b>19</b>
11890	Algorithmen und Berechenbarkeit	20
14360	Einführung in die Technische Informatik	22
14370	Fachstudie Softwaretechnik	24
14390	Programmentwicklung	26
14480	Sichere und zuverlässige Softwaresysteme	28
16500	Software Engineering	29
16510	Software-Praktikum	31
<b>210</b>	<b>Kernmodul Studienprojekt</b>	<b>32</b>
16780	Studienprojekt-Th	33
<b>300</b>	<b>Ergänzungsmodule</b>	<b>35</b>
310	Katalog SWT	36
10120	Grundlagen der Modellbildung und Simulation	37
10150	Grundlagen des Compilerbaus und der Programmiersprachen	39
10210	Mensch-Computer-Interaktion	41
10330	Systemkonzepte und -programmierung	43
16790	Rechnerorganisation 1	45
320	Katalog ISG	47
10030	Architektur von Anwendungssystemen	48
10080	Datenbanken und Informationssysteme	50
10090	Grundlagen der Eingebetteten Systeme	52
10100	Grundlagen der Graphischen Ingenieursysteme	54
10110	Grundlagen der Künstlichen Intelligenz	56
10120	Grundlagen der Modellbildung und Simulation	58
10140	Grundlagen der Rechnerarchitektur	60

**Modulhandbuch Bachelor of Science Softwaretechnik**

---

10150	Grundlagen des Compilerbaus und der Programmiersprachen .....	62
10210	Mensch-Computer-Interaktion .....	64
10220	Modellierung .....	66
10330	Systemkonzepte und -programmierung .....	68
11110	Verteilte Systeme .....	70
16790	Rechnerorganisation 1 .....	72
330	Katalog ISW .....	74
10040	Bildsynthese .....	75
10050	Bildverstehen .....	77
10060	Computergraphik .....	79
10090	Grundlagen der Eingebetteten Systeme .....	81
10100	Grundlagen der Graphischen Ingenieursysteme .....	83
10110	Grundlagen der Künstlichen Intelligenz .....	85
10120	Grundlagen der Modellbildung und Simulation .....	87
10140	Grundlagen der Rechnerarchitektur .....	89
10150	Grundlagen des Compilerbaus und der Programmiersprachen .....	91
10170	Imaging Science .....	93
10180	Information Retrieval und Text Mining .....	95
10210	Mensch-Computer-Interaktion .....	97
10250	Parallele Systeme .....	99
10300	Rechnernetze .....	101
10330	Systemkonzepte und -programmierung .....	103
11330	Visualisierung .....	105
14380	Hardware Verification and Quality Assessment .....	107
16790	Rechnerorganisation 1 .....	109
<b>400</b>	<b>Schlüsselqualifikationen fachaffin .....</b>	<b>111</b>
16610	Studienprojekt-Pr .....	112
<b>800</b>	<b>Schlüsselqualifikationen fachübergreifend gesamt .....</b>	<b>114</b>
900	Schlüsselqualifikationen fachübergreifend .....	115
901	Kompetenzbereich 1: Methodische Kompetenzen .....	116
902	Kompetenzbereich 2: Soziale Kompetenzen .....	117
903	Kompetenzbereich 3: Kommunikative Kompetenzen .....	118
904	Kompetenzbereich 4: Personale Kompetenzen .....	119
905	Kompetenzbereich 5: Recht, Wirtschaft, Politik .....	120
930340	English for Software Engineering .....	121
<b>10160</b>	<b>Hardwarebeschreibungssprachen .....</b>	<b>122</b>
<b>11900</b>	<b>Design and Test of Systems on a Chip .....</b>	<b>123</b>



## **Modul 100 Basismodule**

zugeordnet zu: Studiengang

---

Zugeordnete Module:	10190	Mathematik für Informatiker und Softwaretechniker
	10260	Programmierkurs
	10280	Programmierung und Software-Entwicklung
	10940	Theoretische Grundlagen der Informatik
	12060	Datenstrukturen und Algorithmen
	16520	Software-Qualität
	17210	Einführung in die Softwaretechnik

---

---

**Modul 10190 Mathematik für Informatiker und Softwaretechniker**

Studiengang:	[909]	Modulkürzel:	080300100
Leistungspunkte:	18.0	SWS:	12.0
Moduldauer:	2 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Wolfgang Rump

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Basismodul, Fachaffine SQ, 1. Semester
- Bachelor Softwaretechnik, Basismodul, Fachaffine SQ, 1. Semester

Das Modul besteht aus zwei Veranstaltungen.

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.

Inhalt:

1. Semester:

- Grundlagen (Aussagenlogik, Mengen, Relationen, Abbildungen, Zahlenmengen, Grundbegriffe der Algebra)
- Lineare Algebra (Vektorräume, lineare Abbildungen, Matrizen, Determinanten, lineare Gleichungssysteme, Eigenwerte, Normalformen, Hauptachsentransformation, Skalarprodukte)
- Analysis (Konvergenz, Zahlenfolgen und Zahlenreihen, stetige Abbildungen, Folgen und Reihen von Funktionen, spezielle Funktionen)

2. Semester:

- Differential- und Integralrechnung (Funktionen einer und mehrerer Variablen, Ableitungen, Taylorentwicklungen, Extremwerte, Integration, Anwendungen)
- Gewöhnliche Differentialgleichungen (elementar lösbare Differentialgleichungen, Existenz und Eindeutigkeit von Lösungen)

Literatur / Lernmaterialien:

- Anna Sändig, Mathematik, Vorlesungskripte, SS 2007
- D. Hachenberger, Mathematik für Informatiker, 2005
- M. Brill, Mathematik für Informatiker, 2001
- P.Hartmann, Mathematik für Informatiker, 2002



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 101901 Vorlesung Mathematik 1 für Informatik und Softwaretechnik
- 101902 Übung Mathematik 1 für Informatik und Softwaretechnik
- 101903 Vorlesung Mathematik 2 für Informatik und Softwaretechnik
- 101904 Übung Mathematik 2 für Informatik und Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 126 Stunden  
Nachbearbeitungszeit: 414 Stunden

Studienleistungen:

Ein Übungsschein aus den beiden Veranstaltungen, jeweils im 1. oder 2. Fachsemester zu erwerben.

Prüfungsleistungen:

Schriftliche Prüfung: Zweistündige Klausur.

Prüfungsnummer/n und  
-name:

- 10191 Mathematik für Informatiker und Softwaretechniker

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Technikpädagogik

**Modul 10260 Programmierkurs**

Studiengang:	[909]	Modulkürzel:	051520010
Leistungspunkte:	3.0	SWS:	2.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Jochen Ludewig

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Basismodul, 1. Semester
- Bachelor Softwaretechnik, Basismodul, 1. Semester

Lernziele:

Selbstständiges Erstellen von Programmen und Lösung von Programmieraufgaben in einer vorgegebenen Programmiersprache wie ADA.

Inhalt:

Der Programmierkurs soll die Vorlesung "Programmierung und Software-Entwicklung" (PSE) ergänzen. Die Teilnehmer erlernen eine weitere Programmiersprache; derzeit ist das Java. Durch Gegenüberstellung zur Sprache, die in PSE gelehrt wird (derzeit Ada), wird die Syntax der neuen Sprache eingeführt. Dabei werden auch die Konzepte der objektorientierten Programmierung vermittelt.

Intensiv betreute praktische Übungen bereiten die Teilnehmer auf die Bearbeitung der Schein-Aufgabe vor.

Literatur / Lernmaterialien:

Lehrveranstaltungen und  
-formen:

- 102601 Übung Programmierkurs

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 21 Stunden  
Nachbearbeitungszeit: 69 Stunden

Studienleistungen:

USL (Übungsschein - Scheinkriterien werden zu Beginn der Veranstaltung angekündigt.)

Prüfungsnummer/n und  
-name:

- 10261 Programmierkurs



Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung
- B.Sc. Technikpädagogik
- BA (Komb) Informatik
- M.Sc. Technikpädagogik

**Modul 10280 Programmierung und Software-Entwicklung**

Studiengang:	[909]	Modulkürzel:	051520005
Leistungspunkte:	9.0	SWS:	6.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Jochen Ludewig

Dozenten: • Bernhard Mitschang

Verwendbarkeit / Zuordnung zum Curriculum: • Bachelor Informatik, Basismodul, 1. Semester  
• Bachelor Softwaretechnik, Basismodul, 1. Semester

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.

Inhalt: • Algorithmen und funktionale Programmierung Kap. 1 verwendet nur die funktionale Teilmenge der Programmiersprache Ada, keine Variablen, keine Prozeduren. Grammatik, Formale Sprachen und BNF werden eingeführt.  
• Imperative Programmierung Kap. 2 erweitert die verwendete Sprache durch die prozeduralen Konzepte, also Variablen und Prozeduren. Zu den Sprachkonstrukten werden Vor- und Nachbedingungen, mit den Schleifen die Invarianten eingeführt. Datentypen werden schrittweise ausgebaut. In Zusammenhang mit den Zeigern werden die Konzepte für Keller und Halde vermittelt. Die Entwicklung einfacher Programme wird gezeigt und geübt.  
• Aufbau und Organisation komplexer Programme. Die Modularisierung, die bei größeren Programmen notwendig ist, führt zur Kapselung und zu den abstrakten Datentypen. Damit entsteht die Möglichkeit, neue Datenstrukturen und Datentypen sicher zu definieren. Die Konzepte der Kompilation und der Interpretation werden erläutert. Wichtige Beispiele komplexer Datentypen werden entwickelt. Die Konzepte der Generalisierung (generische Einheiten) werden vermittelt.



## Modulhandbuch Bachelor of Science Softwaretechnik

- Ausnahmebehandlung Möglichkeiten und Probleme der Ausnahmebehandlung sind Gegenstände dieses kurzen Kapitels.
- Objektorientierte Programmierung Am Ende des Semesters steht ein Ausblick in die objektorientierte Programmierung, d.h. die Umsetzung der bereits bekannten Konzepte (ADTs) in die objektorientierte Sichtweise und die Vererbung. Dieser Teil bereitet die Programmierung in einer objektorientierten Sprache (3. Semester) vor.

Literatur / Lernmaterialien:

Manuskripte: V.Claus (WS 08/09 bis SS 2009)

Appelrath, Hans-Jürgen und Ludewig, Jochen, "Skriptum Informatik - eine konventionelle Einführung", Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999

Nagl., M., "Softwaretechnik mit Ada 95. Entwicklung großer Systeme.", Vieweg-Verlag, Wiesbaden 1999

Barnes, J.G.P., "Programming in Ada 95", 2. Auflage, Addison-Wesley 1998

Lehrveranstaltungen und -formen:

- 102801 Vorlesung Programmierung und Softwareentwicklung
- 102802 Übung Programmierung und Softwareentwicklung

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 63 Stunden  
Vor-/Nachbearbeitungszeit: 187 Stunden

Prüfungsvorbereitung: 20 Stunden

Studienleistungen:

Studienleistung: Übungsschein, Vor. 3 mal vortragen in den Übungen und mindestens 50% der Übungspunkte erwerben, Teilnahme an den Zwischenklausuren.

Prüfungsleistungen:

Modulprüfung: Schriftliche Prüfung von 120 Minuten Dauer

Prüfungsnummer/n und -name:

- 10281 Programmierung und Software-Entwicklung

Exportiert durch:



## Modulhandbuch Bachelor of Science Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Mechatronik
- B.Sc. Maschinelle Sprachverarbeitung
- B.Sc. Technikpädagogik
- B.Sc. Simulation Technology
- BA (Komb) Informatik
- M.Sc. Technikpädagogik

**Modul 10940 Theoretische Grundlagen der Informatik**

Studiengang:	[909]	Modulkürzel:	050420005
Leistungspunkte:	12.0	SWS:	8.0
Moduldauer:	2 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Volker Diekert

Dozenten:

- Ulrich Hertrampf
- Volker Diekert

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Basismodul, 1. Semester
- Bachelor Softwaretechnik, Basismodul, 1. Semester

Das Modul Theoretische Grundlagen der Informatik besteht aus den beiden Veranstaltungen Logik und Diskrete Strukturen sowie Automaten und Formale Sprachen.

Lernziele:

- 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.

- 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.

Inhalt:

- 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.

- Automaten und Formale Sprachen:

Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten,



## Modulhandbuch Bachelor of Science Softwaretechnik

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.

Literatur / Lernmaterialien:

- John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988
- Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999

Lehrveranstaltungen und -formen:

- 109401 Vorlesung Logik und Diskrete Strukturen
- 109402 Übung Logik und Diskrete Strukturen
- 109403 Vorlesung Automaten und Formale Sprachen
- 109404 Übung Automaten und Formale Sprachen

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 84 Stunden  
Nachbearbeitungszeit: 276 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Schriftliche Prüfung von 120 Minuten Dauer.

Prüfungsnummer/n und -name:

- 10941 Theoretische Grundlagen der Informatik

Studiengänge die dieses Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Maschinelle Sprachverarbeitung
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik

**Modul 12060 Datenstrukturen und Algorithmen**

Studiengang:	[909]	Modulkürzel:	051510005
Leistungspunkte:	9.0	SWS:	6.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Erhard Plödereder

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Basismodul, 2. Semester
- Bachelor Softwaretechnik, Basismodul, 2. Semester

Teilnahme an den Übungen ist Pflicht. Grundsätzlich gelten folgende Regeln: In den Übungen muss jeder Student und jede Studentin drei Mal vorgetragen haben und mindestens 50% der Übungspunkte erwerben. Im Rahmen der Übungen finden auch bepunktete Zwischenklausuren statt. Die in den Übungen und den Zwischenklausuren erworbenen Punkte werden zu 25% auf die Endnote angerechnet.

Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden jährlich geprüft, geeignet angepasst und zu Beginn der Lehrveranstaltung bekannt gegeben.

Lernziele:

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.

Konkret:

- Kenntnis der Eigenschaften elementarer und häufig benötigter Algorithmen
- Verständnis für die Auswirkungen theoretischer und tatsächlicher Komplexität
- Erweiterung der Kompetenz im Entwurf und Verstehen von Algorithmen und der zugehörigen Datenstrukturen
- Erste Begegnung mit nebenläufigen Algorithmen; sowohl „originär“ parallel, als auch parallelisierte Versionen bereits vorgestellter sequentiell

Inhalt:

- Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen
- Komplexität und Effizienz von Algorithmen, O-Notation



## Modulhandbuch Bachelor of Science Softwaretechnik

---

	<ul style="list-style-type: none"><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>
Literatur / Lernmaterialien:	<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>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 120601 Vorlesung Datenstrukturen und Algorithmen</li><li>• 120602 Übung Datenstrukturen und Algorithmen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden
Studienleistungen:	Prüfungsvorleistung: Übungsschein.
Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 12061 Datenstrukturen und Algorithmen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Mathematik</li><li>• B.Sc. Softwaretechnik</li><li>• B.Sc. Wirtschaftsinformatik</li><li>• B.Sc. Mechatronik</li><li>• B.Sc. Maschinelle Sprachverarbeitung</li><li>• B.Sc. Technikpädagogik</li><li>• BA (Komb) Informatik</li><li>• M.Sc. Technikpädagogik</li></ul>

**Modul 16520 Software-Qualität**

Studiengang:	[909]	Modulkürzel:	051520105
Leistungspunkte:	3.0	SWS:	2.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Jochen Ludewig

Dozenten:

- Jochen Ludewig
- Ivan Bogicevic

Verwendbarkeit / Zuordnung zum Curriculum:

- Bachelor Softwaretechnik, Basismodul, 1. Semester

Lernziele:

Die Teilnehmer kennen und verstehen den Begriff der Software-Qualität. Sie kennen Techniken, deren Anwendung zu einer guten Software-Qualität beiträgt, und können sie anwenden.

Inhalt:

In der Vorlesung wird der Begriff der Software-Qualität vermittelt und am Beispiel anschaulich gemacht.  
In der Übung wird ein großes Softwaresystem bearbeitet.

Literatur / Lernmaterialien:

- Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2007

Lehrveranstaltungen und -formen:

- 165201 Vorlesung Software-Qualität
- 165202 Übung Software-Qualität

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 21 Stunden  
Nachbearbeitungszeit: 69 Stunden

Studienleistungen:

- USL (Übungsschein; Scheinkriterien werden zu Beginn der Veranstaltung angekündigt.)

Prüfungsleistungen:

Keine

Prüfungsnummer/n und -name:

- 16521 Software-Qualität



## Modulhandbuch Bachelor of Science Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik

**Modul 17210 Einführung in die Softwaretechnik**

Studiengang:	[909]	Modulkürzel:	051520015
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Jochen Ludewig

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Kernmodul, 4. Semester
- Bachelor Softwaretechnik, Kernmodul, 2. Semester

EST ist, wie der Name sagt, die allgemeine Einführung in die Softwaretechnik. Sie kommt auch, wenn die Voraussetzungen erfüllt sind, für andere Fachrichtungen als Softwaretechnik und Informatik in Frage.

Lernziele:

Die Teilnehmer kennen die Grundbegriffe der Softwaretechnik und haben wichtige Techniken des Softwareprojekt-Managements und der Software-Entwicklung erlernt.

Inhalt:

Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung, wie sie in der Praxis stattfindet. Die einzelnen Themen sind:

- Abgrenzung und Motivation des Software Engineerings
- Vorgehensmodelle; Software-Management; Software-Prüfung und Qualitätssicherung
- Methoden, Sprachen und Werkzeuge für die einzelnen Phasen:

Spezifikation, Grobentwurf, Feinentwurf, Codierung, Test

Lehrveranstaltungen und  
-formen:

- 172101 Vorlesung Einführung in die Softwaretechnik
- 172102 Übung Einführung in die Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 17211 Einführung in die Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Mechatronik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik



## **Modul 200 Kernmodule**

zugeordnet zu: Studiengang

---

Zugeordnete Module:	11890	Algorithmen und Berechenbarkeit
	14360	Einführung in die Technische Informatik
	14370	Fachstudie Softwaretechnik
	14390	Programmentwicklung
	14480	Sichere und zuverlässige Softwaresysteme
	16500	Software Engineering
	16510	Software-Praktikum

---

---

**Modul 11890 Algorithmen und Berechenbarkeit**

---

Studiengang:	[909]	Modulkürzel:	050420020
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	

---

---

Dozenten:

- Ulrich Hertrampf
- Volker Diekert

Verwendbarkeit / Zuordnung zum Curriculum: Bachelor Softwaretechnik, Kernmodul, 3. Semester  
Die Veranstaltung ist nicht im Bachelor Informatik anrechenbar.

Lernziele: Die Studierenden kennen die Klassifizierung von Algorithmen in effizient berechenbar, NP-vollständig, PSPACE-Algorithmen und prinzipielle Unberechenbarkeit. Sie haben wichtige Entwurfstrategien und Analysemethoden kennengelernt.

Inhalt: Berechenbarkeit vs. Unberechenbarkeit, Church These, NP-Vollständigkeit, PSPACE-Algorithmen (QBF).  
Entwurfstrategien: Teile und Beherrsche, gierig (greedy), Dynamisches Programmieren, Randomisierte Algorithmen

Literatur / Lernmaterialien:

- John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms (Second Edition), 2001
- Volker Diekert, Entwurf und Analyse effizienter Algorithmen (Vorlesungsskript), 2006

Lehrveranstaltungen und -formen:

- 118901 Vorlesung Algorithmen und Berechenbarkeit
- 118902 Übung Algorithmen und Berechenbarkeit

Abschätzung Arbeitsaufwand: Präsenzzeit: 42 h Selbststudiumszeit / Nacharbeitszeit: 138 h  
Gesamt: 180 h

Studienleistungen: Prüfungsvorleistung: Übungsschein.



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsleistungen: Schriftliche Prüfung von 90 Minuten Dauer.

Prüfungsnummer/n und  
-name: • 11891 Algorithmen und Berechenbarkeit

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik



# Modulhandbuch Bachelor of Science Softwaretechnik

## Modul 14360 Einführung in die Technische Informatik

Studiengang:	[909]	Modulkürzel:	051400105
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

Verständnis für elektrische Bauelemente und Komponenten von Computer-Systemen, Analyse und Entwurf digitaler Schaltungen.

Inhalt:

- Elektrische Grundgrößen
- Ohmsches Gesetz
- Kirchhoffsche Gesetze
- Widerstand
- Kondensator
- CMOS-Technologie
- Logik- und Speicherschaltungen
- Boolesche Algebra
- Schaltalgebra
- Entwurf und Minimierung von Schaltfunktionen
- Rückkopplung, Zustandsbegriff
- Flipflops
- Implementierung einfacher endlicher Automaten
- Verzögerungsanalyse
- Datenpfadelemente
- Informationsdarstellung
- Rechnergrundstrukturen
- Operationswerke
- Speicherorganisation

Literatur / Lernmaterialien:

Literatur, die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben.

Lehrveranstaltungen und  
-formen:

- 143601 Vorlesung Einführung in die Technische Informatik
- 143602 Gruppenübungen Einführung in die Technische Informatik



## Modulhandbuch Bachelor of Science Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 126 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Klausur 90 min

Prüfungsnummer/n und  
-name:

- 14361 Einführung in die Technische Informatik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 14370 Fachstudie Softwaretechnik**

Studiengang:	[909]	Modulkürzel:	051520185
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 6. Semester

Lernziele:

Die Teilnehmer sind in der Lage, eine konkrete praktische Frage der Softwaretechnik, beispielsweise über die anzuwendende Methode oder das geeignete Werkzeug, zu analysieren und zu entscheiden und ihre Entscheidung angemessen zu präsentieren. Die Arbeit erfolgt in Dreiergruppen.

Inhalt:

Eine Gruppe analysiert eine (im Allgemeinen aus der Praxis kommende) Frage auf der Basis der Literatur und eigener Untersuchungen, auch Befragungen, und präsentiert ihre Empfehlung mündlich und in Form eines Berichts.

Lehrveranstaltungen und  
-formen:

- 143701 Praktikum Fachstudie Softwaretechnik
- 143702 Teamarbeit an den beteiligten Instituten mit örtlicher fachlicher Betreuung

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 126 Stunden

Studienleistungen:

- unbenoteter Schein, USL

Prüfungsleistungen:

Keine.

Prüfungsnummer/n und  
-name:

- 14371 Fachstudie Softwaretechnik



## Modulhandbuch Bachelor of Science Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik

**Modul 14390 Programmentwicklung**

Studiengang:	[909]	Modulkürzel:	051520120
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

Die Teilnehmer kennen die Prinzipien der objektorientierten Programmierung und sind in der Lage, Programme in UML zu beschreiben und in Java zu implementieren.

Inhalt:

- Grundlagen der objektorientierten Programmierung
- Spezifikation und Entwurf objektorientierter Programme mit UML
- Programmierung in Java

Literatur / Lernmaterialien:

- Rumbaugh, Jacobson, Booch, The unified modeling language reference manual, 2nd ed., 2004
- Rupp, Queins, Zengler, UML 2 glasklar: Praxiswissen für die UML-Modellierung, 3. Aufl. 2007

Lehrveranstaltungen und  
-formen:

- 143901 Vorlesung Programmentwicklung
- 143902 Übung Programmentwicklung

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 126 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Klausur 60 min

Prüfungsnummer/n und  
-name:

- 14391 Programmentwicklung



## Modulhandbuch Bachelor of Science Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik

**Modul 14480 Sichere und zuverlässige Softwaresysteme**

Studiengang:	[909]	Modulkürzel:	051520115
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 5. Semester

Die Lehrveranstaltung ist durch den zukünftigen Inhaber des Lehrstuhls Zuverlässige Softwaresysteme zu gestalten und von diesem anzubieten.

Lernziele:

Die Teilnehmer beherrschen Techniken, deren Anwendung die Zuverlässigkeit der Software verbessert oder garantiert. Dabei stehen formale Ansätze im Vordergrund.

Inhalt:

- Notationen und Verfahren zur formalen Beschreibung und Prüfung der Software

Lehrveranstaltungen und  
-formen:

- 144801 Vorlesung Sichere und zuverlässige Softwaresysteme
- 144802 Übung Sichere und zuverlässige Softwaresysteme

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Klausur 90 min

Prüfungsnummer/n und  
-name:

- 14481 Sichere und zuverlässige Softwaresysteme

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik

**Modul 16500 Software Engineering**

Studiengang:	[909]	Modulkürzel:	051520110
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 4. Semester

Lernziele:

Die Teilnehmer haben tiefe und umfassende Kenntnisse auf dem Gebiet des Softwareprojekt-Managements und in den Techniken der Software-Bearbeitung.

Inhalt:

Ergänzend zur "Einführung in die Softwaretechnik" und daran anknüpfend behandelt diese Lehrveranstaltung folgende Themen:

- Geschichte des Software Engineerings
- Organisationsaspekte der Software-Bearbeitung
- Software-Prozesse, Prozess-Bewertung und -Verbesserung
- Software-Wartung
- Weitere ausgewählte Kapitel des Software Engineerings

Literatur / Lernmaterialien:

- Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2007

Lehrveranstaltungen und  
-formen:

- 165001 Vorlesung Software Engineering
- 165002 Übung Software Engineering

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Klausur 90 min.



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 16501 Software Engineering

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik

**Modul 16510 Software-Praktikum**

Studiengang:	[909]	Modulkürzel:	051520180
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	Jochen Ludewig

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

Die Teilnehmer können eine Software-Entwicklung von der Spezifikation bis zur Auslieferung durchführen.

Inhalt:

Die Teilnehmer bearbeiten in Dreiergruppen eine zentral gestellte Aufgabe. Sie erheben dazu die notwendigen Informationen, erstellen die notwendigen Dokumente und implementieren ein Programm, das die Aufgabe löst.

Literatur / Lernmaterialien:

- Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2007

Lehrveranstaltungen und  
-formen:

- 165101 Praktikum Software-Praktikum

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 126 Stunden

Studienleistungen:

unbenoteter Schein, USL

Prüfungsleistungen:

keine

Prüfungsnummer/n und  
-name:

- 16511 Software-Praktikum

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik



**Modul 210 Kernmodul Studienprojekt**

zugeordnet zu: Studiengang

---

Zugeordnete Module:                      16780    Studienprojekt-Th

---

---

**Modul 16780 Studienprojekt-Th**

Studiengang:	[909]	Modulkürzel:	051520192
Leistungspunkte:	9.0	SWS:	6.0
Moduldauer:	1 Semester	Turnus:	unregelmäßig
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Kernmodul, 4. Semester

Diese Veranstaltung ist der Theorie-Teil des Moduls "Studienprojekt" und entsprechend eng an den Praxisteil gekoppelt. Das Projekt wird mit einer Gesamtnote bewertet, die aus den gewichteten Noten der beiden Teile entsteht. Siehe auch "Studienprojekt".

Lernziele:

Vorlesung und Seminar dienen dazu, theoretische Grundlagen zum Studienprojekt-Pr zu vermitteln und die Arbeit im Projekt zu reflektieren.

Inhalt:

Die konkreten Themen und Inhalte hängen vom jeweiligen Projekt ab.

Literatur / Lernmaterialien:

- Deininger, Lichter, Ludewig, Schneider, Studien-Arbeiten, 5. Aufl. 2005

Lehrveranstaltungen und  
-formen:

- 167801 Vorlesung Studienprojekt-Th
- 167802 Seminar Studienprojekt-Th

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 63 Stunden  
Nachbearbeitungszeit: 189 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Für das Seminar gibt es einen benoteten Schein, für den Vorlesungsteil eine mündliche Prüfung von 25 min Dauer. In der Modulnote wird das Seminar mit 1, die Prüfung mit 2 gewichtet.



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 16781 Studienprojekt-Th - Prüfung
- 16782 Studienprojekt-Th - Schein

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik



## **Modul 300 Ergänzungsmodule**

zugeordnet zu: Studiengang

---

Zugeordnete Module:	310	Katalog SWT
	320	Katalog ISG
	330	Katalog ISW

---

---

**Modul 310 Katalog SWT**

---

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Zugeordnete Module	10120	Grundlagen der Modellbildung und Simulation
	10150	Grundlagen des Compilerbaus und der Programmiersprachen
	10210	Mensch-Computer-Interaktion
	10330	Systemkonzepte und -programmierung
	16790	Rechnerorganisation 1

Dozenten:

Studiengänge die dieses Modul nutzen :

- B.Sc. Elektrotechnik und Informationstechnik
- B.Sc. Technische Biologie
- B.Sc. Softwaretechnik
- B.Sc. Technologiemanagement
- B.Sc. Wirtschaftsinformatik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Erneuerbare Energien
- B.Sc. Architektur und Stadtplanung
- B.Sc. Technikpädagogik
- BA (Komb) Germanistik
- M.Sc. Elektrotechnik und Informationstechnik

**Modul 10120 Grundlagen der Modellbildung und Simulation**

Studiengang:	[909]	Modulkürzel:	051240010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Peter Bastian

Dozenten: • Peter Bastian

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 4. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 4. Semester

Lernziele:

Beherrschung des grundsätzlichen Vorgehens in der Modellbildung, Kenntnis einer Auswahl diskreter und kontinuierlicher Modelle und deren Simulation, Beherrschung der Fehler- und Effizienzanalyse numerischer Verfahren.

Inhalt:

Diese Vorlesung bietet eine Einführung in die Grundlagen der Modellbildung und Simulation mit dem Ziel der Vorbereitung auf weiterführende Vorlesungen in diesem Bereich. Da Simulationsmethoden oft für viele verschiedene Problemklassen einsetzbar sind ist die Vorlesung methodisch strukturiert. Nach einer Einführung des Modellbegriffs wird auf die Grundlagen der ereignisgesteuerten Simulation sowie auf zelluläre Automaten eingegangen. Den Hauptteil der Vorlesung bilden dann kontinuierliche Modelle sowie deren numerische Behandlung. Am Beispiel der Analyse elektrischer Netzwerke werden lineare Gleichungssysteme (direkte Methoden, direkte Methoden für dünnbesetzte Matrizen sowie Krylovraumverfahren), differentiell-algebraische Gleichungen (Existenz von Lösungen und Stabilität, Einschrittverfahren, Schrittweitenkontrolle, steife Probleme) besprochen. Den Abschluss bildet eine kurze Einführung in Eigenschaften und Lösung partieller Differentialgleichungen (Typeinteilung, Finite Differenzen).

Literatur / Lernmaterialien:

- Peter Bastian, Grundlagen der Modellbildung und Simulation, 2008



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 101201 Vorlesung Grundlagen der Modellbildung und Simulation
- 101202 Übung Grundlagen der Modellbildung und Simulation

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer; bei geringer  
Teilnehmerzahl mündliche Prüfung von 30 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10121 Grundlagen der Modellbildung und Simulation

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10150 Grundlagen des Compilerbaus und der Programmiersprachen**

Studiengang:	[909]	Modulkürzel:	051510015
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Erhard Plödereder

Dozenten: • Erhard Plödereder

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 5. Semester

Pflichtvorlesung für die Vertiefungslinie Programmiersprachen und Compilerbau.

Lernziele:

Die Studierenden haben die Grundkenntnisse erlangt, die zur effizienten Verwendung von Lexer- und Parser-Generatoren zur Analyse von Eingabetexten nötig sind. Sie haben gelernt, die Fehlermeldungen aus diesen Generatoren und den Compilern oder Interpretern richtig einzuordnen. Ferner haben sie durch Betrachtung der Implementierungsmodelle typischer Programmiersprachenkonstrukte Verständnis für das Ausführungsverhalten und für typische, gefährliche Fehlerquellen in Anwendungsprogrammen erlangt.

Inhalt:

Compilerarchitekturen im Überblick; lexikalische und syntaktische Analyse von Texten mit formaler Grammatik, insb. von Programmiersprachen. Lexikalische Analyse: endliche Automaten und ihre Implementierung; Syntaxanalyse: diverse Parser-Strategien, ihre Implementierung und Eigenschaften. Methoden der automatischen Generierung von Analysatoren aus Spezifikationen der Grammatiken. Fehlererkennung und -behandlung. Analyse der statischen Semantik: Grundbegriffe und elementare Methoden. Eigenschaften von Programmiersprachen; Realisierung der Laufzeitsemantik prozeduraler Programmiersprachen aus Benutzersicht, insbesondere Implementierungsmodelle der Speicherverwaltung und der Unterprogrammaufrufe. Vermeidung typischer Fehlerquellen und überraschender Probleme in Anwendungsprogrammen.



## Modulhandbuch Bachelor of Science Softwaretechnik

---

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Aho, Sethi, Ullman, Compilers - Principles, Techniques, and Tools, 1988</li><li>• Wilhelm, Maurer, Uebersetzerbau, 1997</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101501 Vorlesung Grundlagen des Compilerbaus und der Programmiersprachen</li><li>• 101502 Übung Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Studienbegleitende Prüfung
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10151 Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10210 Mensch-Computer-Interaktion**

Studiengang:	[909]	Modulkürzel:	051900001
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Thomas Ertl

Dozenten:	<ul style="list-style-type: none"><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li><li>• Bachelor Softwaretechnik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li></ul>
Lernziele:	<p>Verständnis der Methoden und Konzepte der Mensch-Computer-Interaktion, insbesondere der graphisch-interaktiven Systeme:</p> <ul style="list-style-type: none"><li>• Architektur und technische Funktionsweise von GUI-Systemen</li><li>• kognitive Grundlagen und Konsequenzen für die Software-Ergonomie</li><li>• praktische Erfahrung mit der Erstellung von Benutzungsoberflächen mit verschiedenen Programmierschnittstellen</li></ul>
Inhalt:	<ul style="list-style-type: none"><li>• Einführung: Bezug zu anderen Gebieten, historische Entwicklung</li><li>• Menschliche Aspekte: sensorische (insbesondere) visuelle Wahrnehmung, Motorik, Gedächtnis-, Aufmerksamkeits- und Problemlösungsmodelle</li><li>• Computer-Aspekte: Ein/Ausgabegeräte, Display-Architekturen und Event-Verarbeitung, Multimedia-Grundlagen (Vektor-/Rastergraphik, Audio/Video, Farbsysteme), 2D-Graphik (Compositing, Rasterisierung, Linien, Polygone, Text, Bilder, APIs)</li><li>• Interaktionskonzepte und -stile: Geräte- vs. Task-Ebene, Kommandozeile, Menüs, Formulare, Gestik, Spracheingabe, graphische Stile: Direkte Manipulation, WYSIWYG, Icons</li><li>• Fenstersysteme und GUI Toolkits Basisaspekte (Fenstermanagement, Event-Zuordnung), Schichtenaufbau (X, WPF), Widgets/Componenten, Toolkit-Architektur (z.B. Qt, AWT/Swing, XML/Web-basiert), Verwendung von Standardkomponenten</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

- Software Ergonomie, Entwurfsprinzipien: Normen, Regeln (Shneidermann), Style Guides, Modelle (MVC), Metaphern, Entwicklungswerkzeuge
- Evaluation: Experten-Evaluation (Walkthrough, GOMS), Benutzer-Evaluation (Think-aloud, Interviews, Auswertung)
- Spezielle Systeme: mobile Geräte, Virtual/Augmented Reality

Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.

Literatur / Lernmaterialien:

- Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004
- Ben Shneiderman, Catherine Plaisant, Designing the User Interfacs, 2005
- Bernhard Preim, Entwicklung interaktiver Systeme, 1999

Lehrveranstaltungen und -formen:

- 102101 Vorlesung Mensch-Computer-Interaktion
- 102102 Übung Mensch-Computer-Interaktion

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Modulprüfung: Schriftliche Prüfung, 120 Minuten.

Prüfungsnummer/n und -name:

- 10211 Mensch-Computer-Interaktion

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung

**Modul 10330 Systemkonzepte und -programmierung**

Studiengang:	[909]	Modulkürzel:	051200005
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Kurt Rothermel

Dozenten:

- Kurt Rothermel
- Frank Leymann

Verwendbarkeit / Zuordnung zum Curriculum:

- Bachelor Informatik, Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

- Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen
- Verstehen systemnaher Konzepte und Mechanismen
- Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.
- Kann systemnahe Software entwerfen und implementieren.
- Kann nebenläufige Programme entwickeln
- Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.

Inhalt:

Grundlegende Systemstrukturen - und organisationen

- zentrale, verteilte, parallele Systeme
- client/server, Producer/Consumer, P2P, Grid
- Betriebssysteme, Systemplattformen (Middleware), Kerne für eingebettete Systeme

Grundlagen der Rechnernetze

- Grundlegende Netzkonzepte und -architekturen
- Grundlegende Kommunikationsprotokolle und -dienste
- Prozessmanagement
- Prozessbeschreibung, -kontrolle
- Threads

Interprozesskommunikation

- Gemeinsamer Speicher
- Message Passing (Messages, RPC/RMI, Message Queuing, Ereignisse)



## Synchronisation

- Abstraktionen für Shared Memory (Semaphore, Monitore, ...)
- Abstraktionen für Message Passing
- Verklemmungen (Modelle, Behandlung)

## Kausalität und logische Uhren Scheduling

- Uniprozessor
- Multiprozessor
- Realtime

## Schutz und Sicherheit

- grundlegende Konzepte der Sicherheit
- ACLs, Capabilities

## Literatur / Lernmaterialien:

- Literatur, siehe Webseite zur Veranstaltung

## Lehrveranstaltungen und -formen:

- 103301 Vorlesung Systemkonzepte und -programmierung
- 103302 Übung Systemkonzepte und -programmierung

## Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

## Studienleistungen:

Keine

## Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer (Faktor 0.7)  
Benoteter Übungsschein (Faktor 0.3)

## Prüfungsnummer/n und -name:

- 10331 Systemkonzepte und -programmierung
- 10332 Systemkonzepte und -programmierung -  
Übungsschein

## Studiengänge die dieses Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik

**Modul 16790 Rechnerorganisation 1**

Studiengang:	[909]	Modulkürzel:	051700006
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Der erste Teil des Moduls Rechnerorganisation im Studiengang Informatik
- Eigenständiges Modul im Studiengang Softwaretechnik

Lernziele:

- Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung
- Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen

Inhalt:

Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Experimente mit Prozessorsimulatoren vertieft. Im einzelnen werden behandelt:

- Grundstrukturen: Stack-, Akkumulator- und Register-basierende Rechner
- Informationsdarstellung in Hardware: Daten- und Befehlsformate, Fehlererkennung und -korrektur, Befehlssätze und Unterstützung von Hochsprachen.
- MIPS als RISC-Bespiel und seine Assemblerprogrammierung
- Grundelemente und Entwurf auf Register-Transfer-Ebene
- Grundzüge einer Hardware-Beschreibungssprache (VHDL)
- Operationswerke: Multiplikation, Division, Gleitkommaeinheiten
- Steuerwerksentwurf und Mikroprogrammierung
- Befehlszyklus und Unterbrechungen
- Pipelining und statisches Scheduling
- Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung, Segmentierung, TLB, MMU und DMA
- Leistungsbewertung: Maßzahlen und CPI, Benchmarking und einfache Anwendung von Warteschlangen



## Modulhandbuch Bachelor of Science Softwaretechnik

---

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• D. A. Patterson, J. L. Hennessy, Computer Organization: The Hardware-Software Interface, 2004</li><li>• J. P. Hayes, Computer Architecture and Organization, 1998</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 167901 Vorlesung Rechnerorganisation 1</li><li>• 167902 Praktische und theoretische Übungen Rechnerorganisation 1</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Übungsaufgaben, 0.30, Lehrveranstaltungsbegleitende Prüfung Klausurarbeit, 0.70, schriftlich, 60 Minuten
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 16791 Rechnerorganisation 1</li><li>• 16792 Rechnerorganisation 1 - Übungsaufgaben</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Softwaretechnik</li></ul>

**Modul 320 Katalog ISG**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

Zugeordnete Module	10030	Architektur von Anwendungssystemen
	10080	Datenbanken und Informationssysteme
	10090	Grundlagen der Eingebetteten Systeme
	10100	Grundlagen der Graphischen Ingenieursysteme
	10110	Grundlagen der Künstlichen Intelligenz
	10120	Grundlagen der Modellbildung und Simulation
	10140	Grundlagen der Rechnerarchitektur
	10150	Grundlagen des Compilerbaus und der Programmiersprachen
	10210	Mensch-Computer-Interaktion
	10220	Modellierung
	10330	Systemkonzepte und -programmierung
	11110	Verteilte Systeme
	16790	Rechnerorganisation 1

Dozenten:

Studiengänge die dieses Modul nutzen :

- B.Sc. Elektrotechnik und Informationstechnik
- B.Sc. Informatik
- B.Sc. Technische Biologie
- B.Sc. Softwaretechnik
- B.Sc. Technologiemanagement
- B.Sc. Erneuerbare Energien
- B.Sc. Architektur und Stadtplanung
- B.Sc. Technikpädagogik
- BA (Komb) Informatik
- M.Sc. Elektrotechnik und Informationstechnik

**Modul 10030 Architektur von Anwendungssystemen**

Studiengang:	[909]	Modulkürzel:	052010002
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Frank Leymann

Dozenten: • Frank Leymann

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 1, 6. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 1, 6. Semester

Anwendungssysteme sind Kollektionen von Anwendungen, die die Geschäftstätigkeit eines Unternehmens unterstützen. Die Architektur eines solchen Anwendungssystems beschreibt die zugrundeliegenden Bestandteile des Systems "im Grossen" und deren Zusammenwirken. In dieser Architektur sind nicht die konkreten Funktionen im Vordergrund, sondern „das Ganze“, seine Eigenschaften und wie man diese Eigenschaften sicherstellt.

Teilnahme an den Übungen ist Pflicht (es dürfen maximal zwei Übungen versäumt werden).  
Mindestens einmal muss in den Übungen die Lösung einer Aufgabe vorgestellt werden.  
Die Klausur im Anschluss des Semesters muss bestanden werden.

Lernziele:

Am Ende des Moduls ist das Konzept einer Anwendungsarchitektur verstanden. Die Rolle von Middleware im Rahmen von Anwendungsarchitekturen ist klar. Grundsätzliche Strukturen und Muster in Anwendungsarchitekturen sind bekannt. Nicht-funktionale Eigenschaften von Anwendungssystemen und deren Bedeutung sind verstanden.

Inhalt:

- "Anwendungsarchitektur": Terminologie, Rollen & Artefakte
- Datenbanksysteme
- Schichtenarchitekturen (Client-Server Cut, N-Tier)
- API, RPC & Middleware
- Transaktionen
- Nachrichtenorientierung
- Message-Oriented Middleware
- TP Monitore (direct TP, queued TP)
- QoS (high availability, scalability, security,...)



# Modulhandbuch Bachelor of Science Softwaretechnik

---

	<ul style="list-style-type: none"><li>• Komponenten &amp; ihre Beschreibung (EJB &amp; WSDL)</li><li>• Application Server (JEE)</li><li>• Architekturstile (POSA,...)</li><li>• Lose Kopplung &amp; SOA</li><li>• WfMS: Programmieren im Grossen</li><li>• Optional: Lizenzierungen, Portabilität, Deployment, Konfiguration, Installation</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002</li><li>• B. Neubauer, T. Ritter, F. Stoinnski, CORBA Komponenten, 2004</li><li>• F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, Pattern-orientierte Software Architektur - Ein Patternsystem, 1998</li><li>• F. Leymann, D. Roller, Production Workflow, 2000</li><li>• L. Hohmann, Beyond Software Architecture, 2003</li><li>• M. Fowler, Patterns of Enterprise Application Architecture, 2003</li><li>• P. Bernstein, E. Newcomer, Principles of Transaction Processing, 1997</li><li>• S. Conrad, W. Hasselbring, A. Koschel, R. Tritsch, Enterprise Application Integration, 2006</li><li>• S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005</li><li>• W. Emmerich, Konstruktion von verteilten Objekten, 2003</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 100301 Vorlesung Grundlagen der Architektur von Anwendungssystemen</li><li>• 100302 Übung Grundlagen der Architektur von Anwendungssystemen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Prüfungsvorleistung: Übungsschein.
Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10031 Architektur von Anwendungssystemen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10080 Datenbanken und Informationssysteme**

Studiengang:	[909]	Modulkürzel:	051200025
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Bernhard Mitschang

Dozenten:

- Bernhard Mitschang
- Holger Schwarz

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 3. Semester

Diese Vorlesung ist als Einstiegsvorlesung für das Fachgebiet Datenbanksysteme konzipiert. Es wird dabei versucht, das Fachgebiet so gründlich und umfassend darzustellen, wie es für den Datenbankprogrammierer erforderlich und angemessen erscheint.

Lernziele:

Die Studierenden haben die erforderlichen Kenntnisse für Datenbankprogrammierer in angemessenem Umfang erworben.

Inhalt:

Diese Vorlesung ist als Einstiegsvorlesung für das Fachgebiet Datenbanksysteme konzipiert. Es wird dabei versucht, das Fachgebiet so gründlich und umfassend darzustellen, wie es für den Datenbankprogrammierer erforderlich und angemessen erscheint.

Stoffauswahl, -umfang und Detaillierungsgrad wurden deshalb aus der Sicht der Anwendung von Datenbanksystemen getroffen, wozu hauptsächlich das Verständnis von Datenmodellen, der Entwurf von logischen Datenbankstrukturen (DB-Schemata) und der Umgang mit Datenbanksprachen gehören. Weiterhin soll durch Stoffauswahl das Verständnis anderer Vorlesungen, die gewisse Querbezüge und Verbindungen zu Datenbanksystemen aufweisen, erleichtert werden. Dazu gehören zum Beispiel Vorlesungen über Informationssysteme, Systemanalyse, Wissensdarstellung, Expertensysteme, Multimedia-Datenbanksysteme oder Rechnergestützte Ingenieursysteme (CAD/CAM).

Übersicht:

- Informationsmodelle (Entity-Relationship-Modell)
- Datenmodelle (Relationenmodell, Netzwerk- und hierarchisches Datenmodell)



## Modulhandbuch Bachelor of Science Softwaretechnik

- Relationale Anfragesprachen (SQL, Relationenalgebra und Relationenkalkül)
- Logischer DB-Entwurf (Relationensynthese und Normalformen)
- Netzwerk-Datenmodell und Hierarchisches Datenmodell

Stoffauswahl, -umfang und Detaillierungsgrad werden aus der Sicht der Anwendung von Datenbanksystemen getroffen.

### Literatur / Lernmaterialien:

- A. Kemper, A. Eickler, Datenbanksysteme - Eine Einführung, 2004
- C. Date, An Introduction to Database Systems, 2003
- H. Garcia-Molina, J. D. Ullman, J. Widom, Database Systems. The Complete Book, 2003
- R. Elmasri, S. Navathe, Fundamentals of Database Systems, 2003

### Lehrveranstaltungen und -formen:

- 100801 Vorlesung Grundlagen der Datenbanken und Informationssysteme
- 100802 Übung Grundlagen der Datenbanken und Informationssysteme

### Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

### Studienleistungen:

keine

### Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer.

### Prüfungsnummer/n und -name:

- 10081 Datenbanken und Informationssysteme

### Studiengänge die dieses Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Maschinelle Sprachverarbeitung
- BA (Komb) Informatik

**Modul 10090 Grundlagen der Eingebetteten Systeme**

Studiengang:	[909]	Modulkürzel:	051711010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Martin Radetzki

Dozenten:	<ul style="list-style-type: none"><li>• Martin Radetzki</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li></ul>
Lernziele:	Methodische und werkzeugunterstützte Konstruktion von Hardware und Software für auf spezifische Anwendungen hin optimierte eingebettete Systeme.
Inhalt:	Modelle zur Systemspezifikation, Modellierungssprachen und ihre Simulation, System- und Architektursynthese, Allokation von Ressourcen und Bindung von Aufgaben/Operationen, Hardware-Software-Partitionierung, Verfahren zur Ablaufplanung für parallele Architekturen, Optimierungsverfahren, Anwendungsspezifische Prozessoren, On-Chip/Board-Verbindungsnetzwerke.
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• J. Teich, Digitale Hardware/Software-Systeme, 2. Auflage, 2007</li><li>• P. Marwedel, Embedded System Design, 2006</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 100901 Vorlesung Grundlagen der Eingebetteten Systeme</li><li>• 100902 Übung Grundlagen der Eingebetteten Systeme</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine.
Prüfungsleistungen:	Bewertete Rechnerübungen - vorlesungsbegleitend (0.25) Schriftliche Prüfung von 90 Minuten Dauer (0.75)



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 10091 Grundlagen der Eingebetteten Systeme
- 10092 Grundlagen der Eingebetteten Systeme -  
Rechnerübungen

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10100 Grundlagen der Graphischen Ingenieursysteme**

---

Studiengang:	[909]	Modulkürzel:	051400005
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Dieter Roller

---

---

Dozenten:	<ul style="list-style-type: none"><li>•</li><li>• Dieter Roller</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li></ul>
Lernziele:	<ul style="list-style-type: none"><li>• Kenntnis und Verständnis von Modellen bei der Produktentwicklung</li><li>• Grundkenntnisse über die wichtigsten Modellarten, Algorithmen und Datenstrukturen und Techniken für den Datenaustausch</li></ul>
Inhalt:	<p>Inhalte:</p> <ul style="list-style-type: none"><li>• Anforderungen an CAD-Systeme</li><li>• zweidimensionale Modelle</li><li>• dreidimensionale Modelle</li><li>• interaktive Modellerstellung</li><li>• Einführung in die Modifikationstechnik u. parametrische Modellierung</li><li>• Methoden zur Modellmodifikation</li><li>• Grundlagen der parametrischen Modellierung</li><li>• Ansätze und Verfahren zur parametrischen Variantenerzeugung</li><li>• Ausgewählte Anwendungsbeispiele</li><li>• Überblick über weitergehende Modellieransätze</li><li>• Datenverwaltung in CAD</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• D. Roller, CAD - Effiziente Anpassungs- und Variantenkonstruktion, Springer-Verlag</li><li>• Literatur, siehe Webseite zur Veranstaltung</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101001 Vorlesung Grundlagen der Graphischen Ingenieursysteme</li><li>• 101002 Übung Grundlagen der Graphischen Ingenieursysteme</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

keine

Prüfungsleistungen:

Schriftliche Prüfung von 60 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10101 Grundlagen der Graphischen Ingenieursysteme

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10110 Grundlagen der Künstlichen Intelligenz**

Studiengang:	[909]	Modulkürzel:	051900205
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Gunther Heidemann

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester

Lernziele:

Der Student / die Studentin beherrscht die Grundlagen der Künstlichen Intelligenz, kann Probleme der KI selbständig einordnen und mit den erlernten Methoden und Algorithmen bearbeiten.

Inhalt:

- Intelligenz
- Agentenbegriff
- Problemlösen durch Suchen, Suchverfahren
- Probleme mit Rand- und Nebenbedingungen
- Spiele
- Aussagen- und Prädikatenlogik
- Logikbasierte Agenten, Wissensrepräsentation
- Inferenz
- Planen
- Unsicherheit, probabilistisches Schließen
- Probabilistisches Schließen über die Zeit
- Sprachverarbeitung
- Entscheidungstheorie
- Lernen

Literatur / Lernmaterialien:

- G. F. Luger, Künstliche Intelligenz, 2001
- S. Russell, P. Norvig, Künstliche Intelligenz, 2004

Lehrveranstaltungen und  
-formen:

- 101101 Vorlesung Grundlagen der Künstlichen Intelligenz
- 101102 Übung Grundlagen der Künstlichen Intelligenz

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden



## Modulhandbuch Bachelor of Science Softwaretechnik

Studienleistungen:

Prüfungsvorleistung: Übungsschein

Prüfungsleistungen:

Schriftliche Prüfung von 120 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10111 Grundlagen der Künstlichen Intelligenz

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung
- BA (Komb) Informatik

**Modul 10120 Grundlagen der Modellbildung und Simulation**

Studiengang:	[909]	Modulkürzel:	051240010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Peter Bastian

Dozenten:	<ul style="list-style-type: none"><li>• Peter Bastian</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li></ul>
Lernziele:	Beherrschung des grundsätzlichen Vorgehens in der Modellbildung, Kenntnis einer Auswahl diskreter und kontinuierlicher Modelle und deren Simulation, Beherrschung der Fehler- und Effizienzanalyse numerischer Verfahren.
Inhalt:	<p>Diese Vorlesung bietet eine Einführung in die Grundlagen der Modellbildung und Simulation mit dem Ziel der Vorbereitung auf weiterführende Vorlesungen in diesem Bereich. Da Simulationsmethoden oft für viele verschiedene Problemklassen einsetzbar sind ist die Vorlesung methodisch strukturiert. Nach einer Einführung des Modellbegriffs wird auf die Grundlagen der ereignisgesteuerten Simulation sowie auf zelluläre Automaten eingegangen. Den Hauptteil der Vorlesung bilden dann kontinuierliche Modelle sowie deren numerische Behandlung. Am Beispiel der Analyse elektrischer Netzwerke werden lineare Gleichungssysteme (direkte Methoden, direkte Methoden für dünnbesetzte Matrizen sowie Krylovraumverfahren), differentiell-algebraische Gleichungen (Existenz von Lösungen und Stabilität, Einschrittverfahren, Schrittweitenkontrolle, steife Probleme) besprochen. Den Abschluss bildet eine kurze Einführung in Eigenschaften und Lösung partieller Differentialgleichungen (Typeinteilung, Finite Differenzen).</p>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Peter Bastian, Grundlagen der Modellbildung und Simulation, 2008</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 101201 Vorlesung Grundlagen der Modellbildung und Simulation
- 101202 Übung Grundlagen der Modellbildung und Simulation

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer; bei geringer  
Teilnehmerzahl mündliche Prüfung von 30 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10121 Grundlagen der Modellbildung und Simulation

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10140 Grundlagen der Rechnerarchitektur**

---

Studiengang:	[909]	Modulkürzel:	051700010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Hans-Joachim Wunderlich

---

---

Dozenten:	<ul style="list-style-type: none"><li>• Hans-Joachim Wunderlich</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 5. Semester</li></ul>
Lernziele:	<ul style="list-style-type: none"><li>• Grundlegendes Verständnis der Herausforderungen bei Einsatz, Entwurf und Fertigung moderner Prozessoren</li><li>• Kenntnis über den Zusammenhang von Kosten, Performanz und Verlustleistung beim Entwurf von Prozessoren und der Konzeption komplexer Systeme</li><li>• Kenntnis und Bewertung aktueller Architekturkonzepte</li></ul>
Inhalt:	<p>Es werden die klassischen Themen der Rechnerarchitektur als Hardware/Software-Schnittstelle sowie weiterführende technologische Themen behandelt.</p> <p>Dazu gehören insbesondere:</p> <ul style="list-style-type: none"><li>• Technologische Grundlagen: Entwurststile und Fertigungstechnik, Ausbeute, Test und Zuverlässigkeit, Kosten und Qualität, Skalierung.</li><li>• Performanz: Taktfrequenz und Befehle pro Takt, Geschwindigkeitsanalyse und -optimierung.</li><li>• Verlustleistung: Verlustleistungsanalyse, Optimierung von Verlustleistung und Performanz, Verlustleistung und Skalierung.</li><li>• Computerarithmetik: Effiziente Hardwareimplementierung der Grundrechenarten, Hardwareimplementierung spezieller Funktionen wie Exponentialfunktion, Logarithmus und trigonometrische Funktionen, Gleitkommaoperationen und Standards, arithmetische Pipelines und Filter, aktuelle Gleitkommaeinheiten wie SPE des Cell Prozessors oder SPARC.</li><li>• Parallelität auf Befehlsebene: Superskalarrechner, statisches und dynamisches Scheduling, Out-of-Order Ausführung und VLIW-Rechner, Multithreading</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

---

	<ul style="list-style-type: none"><li>• Parallele Architekturen: Shared Memory und Message Passing, Multi-Core Prozessoren und Multi-Core Systeme auf einem Chip.</li><li>• Speicherhierarchie: Speichertechnologie und Cacheentwurf.</li><li>• Fehlertoleranztechniken: Einzelprozessoren und Mehrprozessorsysteme.</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• I. Koren, Computer Arithmetic Algorithms, 2001</li><li>• J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 2006</li><li>• S. Iman, M. Pedram, Logic Synthesis for Low Power VLSI Designs, 1997</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101401 Vorlesung Grundlagen der Rechnerarchitektur</li><li>• 101402 Übung Grundlagen der Rechnerarchitektur</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10141 Grundlagen der Rechnerarchitektur</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10150 Grundlagen des Compilerbaus und der Programmiersprachen**

Studiengang:	[909]	Modulkürzel:	051510015
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Erhard Plödereder

Dozenten: • Erhard Plödereder

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 5. Semester

Pflichtvorlesung für die Vertiefungslinie Programmiersprachen und Compilerbau.

Lernziele:

Die Studierenden haben die Grundkenntnisse erlangt, die zur effizienten Verwendung von Lexer- und Parser-Generatoren zur Analyse von Eingabetexten nötig sind. Sie haben gelernt, die Fehlermeldungen aus diesen Generatoren und den Compilern oder Interpretern richtig einzuordnen. Ferner haben sie durch Betrachtung der Implementierungsmodelle typischer Programmiersprachenkonstrukte Verständnis für das Ausführungsverhalten und für typische, gefährliche Fehlerquellen in Anwendungsprogrammen erlangt.

Inhalt:

Compilerarchitekturen im Überblick; lexikalische und syntaktische Analyse von Texten mit formaler Grammatik, insb. von Programmiersprachen. Lexikalische Analyse: endliche Automaten und ihre Implementierung; Syntaxanalyse: diverse Parser-Strategien, ihre Implementierung und Eigenschaften. Methoden der automatischen Generierung von Analysatoren aus Spezifikationen der Grammatiken. Fehlererkennung und -behandlung. Analyse der statischen Semantik: Grundbegriffe und elementare Methoden. Eigenschaften von Programmiersprachen; Realisierung der Laufzeitsemantik prozeduraler Programmiersprachen aus Benutzersicht, insbesondere Implementierungsmodelle der Speicherverwaltung und der Unterprogrammaufrufe. Vermeidung typischer Fehlerquellen und überraschender Probleme in Anwendungsprogrammen.



## Modulhandbuch Bachelor of Science Softwaretechnik

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Aho, Sethi, Ullman, Compilers - Principles, Techniques, and Tools, 1988</li><li>• Wilhelm, Maurer, Uebersetzerbau, 1997</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101501 Vorlesung Grundlagen des Compilerbaus und der Programmiersprachen</li><li>• 101502 Übung Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Studienbegleitende Prüfung
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10151 Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10210 Mensch-Computer-Interaktion**

Studiengang:	[909]	Modulkürzel:	051900001
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Thomas Ertl

Dozenten:	<ul style="list-style-type: none"><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li><li>• Bachelor Softwaretechnik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li></ul>
Lernziele:	<p>Verständnis der Methoden und Konzepte der Mensch-Computer-Interaktion, insbesondere der graphisch-interaktiven Systeme:</p> <ul style="list-style-type: none"><li>• Architektur und technische Funktionsweise von GUI-Systemen</li><li>• kognitive Grundlagen und Konsequenzen für die Software-Ergonomie</li><li>• praktische Erfahrung mit der Erstellung von Benutzungsoberflächen mit verschiedenen Programmierschnittstellen</li></ul>
Inhalt:	<ul style="list-style-type: none"><li>• Einführung: Bezug zu anderen Gebieten, historische Entwicklung</li><li>• Menschliche Aspekte: sensorische (insbesondere) visuelle Wahrnehmung, Motorik, Gedächtnis-, Aufmerksamkeits- und Problemlösungsmodelle</li><li>• Computer-Aspekte: Ein/Ausgabegeräte, Display-Architekturen und Event-Verarbeitung, Multimedia-Grundlagen (Vektor-/Rastergraphik, Audio/Video, Farbsysteme), 2D-Graphik (Compositing, Rasterisierung, Linien, Polygone, Text, Bilder, APIs)</li><li>• Interaktionskonzepte und -stile: Geräte- vs. Task-Ebene, Kommandozeile, Menüs, Formulare, Gestik, Spracheingabe, graphische Stile: Direkte Manipulation, WYSIWYG, Icons</li><li>• Fenstersysteme und GUI Toolkits Basisaspekte (Fenstermanagement, Event-Zuordnung), Schichtenaufbau (X, WPF), Widgets/Componenten, Toolkit-Architektur (z.B. Qt, AWT/Swing, XML/Web-basiert), Verwendung von Standardkomponenten</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

- Software Ergonomie, Entwurfsprinzipien: Normen, Regeln (Shneidermann), Style Guides, Modelle (MVC), Metaphern, Entwicklungswerkzeuge
- Evaluation: Experten-Evaluation (Walkthrough, GOMS), Benutzer-Evaluation (Think-aloud, Interviews, Auswertung)
- Spezielle Systeme: mobile Geräte, Virtual/Augmented Reality

Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.

Literatur / Lernmaterialien:

- Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004
- Ben Shneiderman, Catherine Plaisant, Designing the User Interfacs, 2005
- Bernhard Preim, Entwicklung interaktiver Systeme, 1999

Lehrveranstaltungen und -formen:

- 102101 Vorlesung Mensch-Computer-Interaktion
- 102102 Übung Mensch-Computer-Interaktion

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Modulprüfung: Schriftliche Prüfung, 120 Minuten.

Prüfungsnummer/n und -name:

- 10211 Mensch-Computer-Interaktion

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung

**Modul 10220 Modellierung**

Studiengang:	[909]	Modulkürzel:	052010001
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Frank Leymann

Dozenten: • Frank Leymann

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Katalog STG, Kernmodul, 4. Semester

Modellierung ist die Basis der Software-Entwicklung. Ein Modell beschreibt z.B. die wesentlichen Strukturen und das Verhalten der zu erstellenden Software, deren Komponenten und die Beziehungen der Komponenten. Ein wichtiger Aspekt ist hierbei auch die Verständlichkeit der Modelle zur Unterstützung der Kommunikation zwischen Projektbeteiligten. Modelle können oft auch transformiert werden, um (iterativ) Artefakte der Ausführungsumgebung zu erhalten.

Teilnahme an den Übungen ist Pflicht (es dürfen maximal zwei Übungen versäumt werden). Mindestens einmal muss in den Übungen die Lösung einer Aufgabe vorgestellt werden. Die Klausur im Anschluss des Semesters muss bestanden werden.

Lernziele:

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.

Inhalt:

- Entity-Relationship Modell & komplexe Objekte
- Relationenmodell & Relationenalgebra, Überblick SQL
- Transformationen von ER nach Relationen
- XML, DTD, XML-Schema, Info-Set, Namensräume, XSLT, XPath
- Metamodelle & RepositoryMDA Konzepte
- RDF, RDF-S & Ontologien
- UML
- Petri Netze, Workflownetze
- BPMN
- IT Landkarten (Modellierung komplexe Systeme - FMC)

Literatur / Lernmaterialien:

- A. Knöpfel, B. Gröne, P. Tabeling, Fundamental Modeling Concepts - Effective Communication of IT Systems, 2005



## Modulhandbuch Bachelor of Science Softwaretechnik

- A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002
- B. Daum, U. Merten, System Architecture With XML, 2003
- M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work - Objektorientierte Modellierung mit UML2, 2005
- P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure, Semantic Web, 2008
- T.J. Teorey, Database Modeling & Design, 2nd Edition,, 1994
- V. Gruhn, D. Pieper, C. Röttgers, MDA, 2006
- W. van der Aalst, K. van Hee, Workflow Management, 2002

Lehrveranstaltungen und  
-formen:

- 102201 Vorlesung Modellierung
- 102202 Übung Modellierung

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer

Prüfungsnummer/n und  
-name:

- 10221 Modellierung

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik

**Modul 10330 Systemkonzepte und -programmierung**

Studiengang:	[909]	Modulkürzel:	051200005
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Kurt Rothermel

Dozenten:

- Kurt Rothermel
- Frank Leymann

Verwendbarkeit / Zuordnung zum Curriculum:

- Bachelor Informatik, Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

- Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen
- Verstehen systemnaher Konzepte und Mechanismen
- Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.
- Kann systemnahe Software entwerfen und implementieren.
- Kann nebenläufige Programme entwickeln
- Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.

Inhalt:

Grundlegende Systemstrukturen - und organisationen

- zentrale, verteilte, parallele Systeme
- client/server, Producer/Consumer, P2P, Grid
- Betriebssysteme, Systemplattformen (Middleware), Kerne für eingebettete Systeme

Grundlagen der Rechnernetze

- Grundlegende Netzkonzepte und -architekturen
- Grundlegende Kommunikationsprotokolle und -dienste
- Prozessmanagement
- Prozessbeschreibung, -kontrolle
- Threads

Interprozesskommunikation

- Gemeinsamer Speicher
- Message Passing (Messages, RPC/RMI, Message Queuing, Ereignisse)



Synchronisation

- Abstraktionen für Shared Memory (Semaphore, Monitore, ...)
- Abstraktionen für Message Passing
- Verklemmungen (Modelle, Behandlung)

Kausalität und logische Uhren Scheduling

- Uniprozessor
- Multiprozessor
- Realtime

Schutz und Sicherheit

- grundlegende Konzepte der Sicherheit
- ACLs, Capabilities

Literatur / Lernmaterialien:

- Literatur, siehe Webseite zur Veranstaltung

Lehrveranstaltungen und -formen:

- 103301 Vorlesung Systemkonzepte und -programmierung
- 103302 Übung Systemkonzepte und -programmierung

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer (Faktor 0.7)  
Benoteter Übungsschein (Faktor 0.3)

Prüfungsnummer/n und -name:

- 10331 Systemkonzepte und -programmierung
- 10332 Systemkonzepte und -programmierung -  
Übungsschein

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik

**Modul 11110 Verteilte Systeme**

Studiengang:	[909]	Modulkürzel:	051200015
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Kurt Rothermel

Dozenten:	<ul style="list-style-type: none"><li>• Kurt Rothermel</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li></ul>
Lernziele:	<ul style="list-style-type: none"><li>• Verstehen der grundsätzlichen Eigenschaften, Konzepte und Verfahren verteilter Systeme.</li><li>• Kann existierende verteilte Anwendungen und Systemplattformen hinsichtlich ihrer Eigenschaften analysieren und verstehen.</li><li>• Kann verteilte Anwendungen/Systemplattformen auf der Grundlage der erlernten Methoden realisieren.</li><li>• Kann sich mit Experten anderer Fachdisziplinen über die Anwendung verteilter Systeme verständigen.</li></ul>
Inhalt:	<ul style="list-style-type: none"><li>• Einführung in die verteilten Systeme</li><li>• Systemmodelle</li><li>• Kommunikation: Nachrichten, Remote Procedure Call (RPC), Remote Method Invocation (RMI)</li><li>• Namensgebung: Generierung und Resolution</li><li>• Zeit und Uhren in verteilten Systemen: Anwendungen, logische Uhren, physikalische Uhren, Uhrensynchronisation</li><li>• Prozesssynchronisation: Wechselseitiger Ausschluß</li><li>• Globaler Zustand: Konzepte, Snapshot Algorithmus, verteiltes Debugging</li><li>• Transaktionsmanagement: Serialisierbarkeit, Sperrverfahren, 2-Phasen-Commit-Protokolle</li><li>• Datenreplikation: Primary Copy, Consensus-Protokolle und andere Algorithmen</li><li>• Sicherheit: Verfahren zur Geheimhaltung, Integrität, Authentifikation und Autorisierung</li><li>• Broadcast-Algorithmen: Verarbeitungsmodell, Broadcast-Semantiken und -Algorithmen</li></ul>
Literatur / Lernmaterialien:	Literatur, siehe Webseite zur Veranstaltung



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 111101 Vorlesung Verteilte Systeme
- 111102 Übungen Verteilte Systeme

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 h  
Selbststudiumszeit / Nachbearbeitungszeit: 138 h  
Gesamt: 180 h

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftlich Prüfung von 90 Minuten Dauer (0.7)  
Benotete Übungen - studienbegleitend (0.3)

Prüfungsnummer/n und  
-name:

- 11111 Verteilte Systeme
- 11112 Verteilte Systeme - Übungen

Exportiert durch:

Institut für Parallele und Verteilte Systeme

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- BA (Komb) Informatik

**Modul 16790 Rechnerorganisation 1**

Studiengang:	[909]	Modulkürzel:	051700006
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Der erste Teil des Moduls Rechnerorganisation im Studiengang Informatik
- Eigenständiges Modul im Studiengang Softwaretechnik

Lernziele:

- Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung
- Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen

Inhalt:

Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Experimente mit Prozessorsimulatoren vertieft. Im einzelnen werden behandelt:

- Grundstrukturen: Stack-, Akkumulator- und Register-basierende Rechner
- Informationsdarstellung in Hardware: Daten- und Befehlsformate, Fehlererkennung und -korrektur, Befehlssätze und Unterstützung von Hochsprachen.
- MIPS als RISC-Bespiel und seine Assemblerprogrammierung
- Grundelemente und Entwurf auf Register-Transfer-Ebene
- Grundzüge einer Hardware-Beschreibungssprache (VHDL)
- Operationswerke: Multiplikation, Division, Gleitkommaeinheiten
- Steuerwerksentwurf und Mikroprogrammierung
- Befehlszyklus und Unterbrechungen
- Pipelining und statisches Scheduling
- Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung, Segmentierung, TLB, MMU und DMA
- Leistungsbewertung: Maßzahlen und CPI, Benchmarking und einfache Anwendung von Warteschlangen



## Modulhandbuch Bachelor of Science Softwaretechnik

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• D. A. Patterson, J. L. Hennessy, Computer Organization: The Hardware-Software Interface, 2004</li><li>• J. P. Hayes, Computer Architecture and Organization, 1998</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 167901 Vorlesung Rechnerorganisation 1</li><li>• 167902 Praktische und theoretische Übungen Rechnerorganisation 1</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Übungsaufgaben, 0.30, Lehrveranstaltungsbegleitende Prüfung Klausurarbeit, 0.70, schriftlich, 60 Minuten
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 16791 Rechnerorganisation 1</li><li>• 16792 Rechnerorganisation 1 - Übungsaufgaben</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Softwaretechnik</li></ul>

**Modul 330 Katalog ISW**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

## Zugeordnete Module

10040	Bildsynthese
10050	Bildverstehen
10060	Computergraphik
10090	Grundlagen der Eingebetteten Systeme
10100	Grundlagen der Graphischen Ingenieursysteme
10110	Grundlagen der Künstlichen Intelligenz
10120	Grundlagen der Modellbildung und Simulation
10140	Grundlagen der Rechnerarchitektur
10150	Grundlagen des Compilerbaus und der Programmiersprachen
10170	Imaging Science
10180	Information Retrieval und Text Mining
10210	Mensch-Computer-Interaktion
10250	Parallele Systeme
10300	Rechnernetze
10330	Systemkonzepte und -programmierung
11330	Visualisierung
14380	Hardware Verification and Quality Assessment
16790	Rechnerorganisation 1

## Dozenten:

Studiengänge die dieses Modul nutzen :

- B.Sc. Elektrotechnik und Informationstechnik
- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Architektur und Stadtplanung
- BA (Komb) Informatik
- M.Sc. Elektrotechnik und Informationstechnik

**Modul 10040 Bildsynthese**

---

Studiengang:	[909]	Modulkürzel:	051900012
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Carsten Dachsbacher

---

## Dozenten:

- Thomas Ertl
- Daniel Weiskopf
- Carsten Dachsbacher

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Katalog Wahl-INF 1, 6. Semester
- Bachelor Softwaretechnik, Katalog Wahl-INF 1, 6. Semester

## Lernziele:

Die Studierenden besitzen Wissen über verschiedene Ansätze und Algorithmen der dreidimensionalen Computergraphik, physikalisch-basierte Verfahren wie Raytracing und Radiosity, die den Lichttransport und die Wechselwirkung mit Materie modellieren, und numerische Methoden wie Monte-Carlo-Integration und Finite-Elemente-Verfahren die es erlauben, die Rendering-Gleichung zu lösen. Interaktive Verfahren nutzen spezielle Eigenschaften moderner Graphikhardware, um mit Hilfe mehrdimensionaler Texturen und anderer Rasterisierungsoperationen realistische Beleuchtungseffekte in Echtzeit zu generieren. Bild-basierte Verfahren verzichten auf eine geometrische Repräsentation der Szene und erzeugen neue Ansichten aus anderen aufgenommenen Bildern.

## Inhalt:

In dieser Vorlesung werden die folgenden Themen behandelt:

- Grafik Hardware und APIs, OpenGL
- Texturen, prozedurale Modelle
- Schattenberechnungen
- Szenengraphen, Culling, Level-of-Detail Verfahren
- Physikalisch-basierte Beleuchtungsberechnung, Fotorealistische Bildsynthese
- Lokale Beleuchtungsmodelle
- Raytracing, Monte-Carlo Methoden
- Radiosity

## Literatur / Lernmaterialien:

- Andrew S. Glassner, Principles of Digital Image Synthesis, 1995



## Modulhandbuch Bachelor of Science Softwaretechnik

- D. Eberly, 3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics, 2000
- J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990
- Literatur, siehe Webseite zur Veranstaltung
- P. Dutre, P. Bekaert, K. Bala, Advanced Global Illumination, 2003
- Tomas Akenine-Möller, Eric Haines, Real-Time Rendering, 2002

Lehrveranstaltungen und  
-formen:

- 100401 Vorlesung Bildsynthese
- 100402 Übung Bildsynthese

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Mündliche Prüfung von 30 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10041 Bildsynthese

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10050 Bildverstehen**

Studiengang:	[909]	Modulkürzel:	051200035
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Paul Levi

## Dozenten:

- Paul Levi
- Viktor Avrutin

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog Wahl-INF 2, 4. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog Wahl-INF 2, 4. Semester

## Lernziele:

Die Teilnehmer kennen und verstehen die Grundlagen der klassischen, verkörperten und verteilten Künstlichen Intelligenz. Ihnen wurden die dabei verwendeten Grundbegriffe so vorgestellt, dass sie als Bausteine von Architekturen intelligenter Systeme betrachtet werden. Dies bedeutet im Einzelnen, dass die Zuhörer am Beispiel des Bildverstehens die Wirkungsweise kognitiver Wahrnehmungsfähigkeiten kennengelernt haben, verstehen wie diese Fähigkeiten im Rahmen des Beobachtungs-, Planungs- und Aktionszyklus in einem Agenten eingesetzt werden und wie diese Fähigkeiten erweitert werden müssen, damit einzelne Agenten sich in einem Team kooperativ (Multiagenten-Systeme) verhalten können.

Die Studierenden sind in der Lage, nicht nur einzelne wesentliche Algorithmen der Künstlichen Intelligenz (einschließlich des Bildverstehens) zu verstehen, sondern auch die wechselseitigen Beziehungen von verschiedenen Methoden zu berücksichtigen, um eine Beurteilung der Konzeption und der Wirkungsweise von intelligenten (kognitiven) technischen Systemen selbst vornehmen zu können.

## Inhalt:

- Einleitung in Problemstellungen und Definitionen
- Einführung in die Grundlagen der Bildverarbeitung
- (Künstliche) Neuronale Netze
- Bedingungsausbreitung (Constraints und ihre Propagierung)
- Probabilistische Inferenz mit Bayes-Netzwerke
- Verteilte Künstliche Intelligenz (VKI)
- Multiagentensysteme (MAS)



## Modulhandbuch Bachelor of Science Softwaretechnik

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• G. F. Luger, Künstliche Intelligenz, 2001</li><li>• Jähne, Bernd, Digitale Bildverarbeitung, 2005</li><li>• Literatur, siehe Webseite zur Veranstaltung</li><li>• S. Russell, P. Norvig, Künstliche Intelligenz, 2004</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 100501 Vorlesung Bildverstehen</li><li>• 100502 Übung Bildverstehen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	keine
Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10051 Bildverstehen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10060 Computergraphik**

Studiengang:	[909]	Modulkürzel:	051900002
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Daniel Weiskopf

Dozenten:	<ul style="list-style-type: none"><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester</li></ul>
Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.
Inhalt:	<p>Folgende Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"><li>• Überblick über den Bildsyntheseprozess</li><li>• Grundlegende Rastergraphik</li><li>• 2D und 3D Geometrietransformationen, 3D Projektion</li><li>• Polygonale und hierarchische Modelle</li><li>• Verdeckungsberechnung</li><li>• Grundlegende Renderingtechniken (Rasterung, Raytracing)</li><li>• Beleuchtungsmodelle</li><li>• Texturen</li><li>• Grundlagen der geometrischen Modellierung (Kurven, Flächen)</li></ul> <p>Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.</p>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• J. Encarnacao, W. Strasser, R. Klein, Graphische Datenverarbeitung (Band1 und 2), 1997</li><li>• J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 100601 Vorlesung Computergraphik
- 100602 Übung Computergraphik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Schriftliche Prüfung von 120 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10061 Computergraphik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- B.Sc. Maschinelle Sprachverarbeitung
- BA (Komb) Informatik

**Modul 10090 Grundlagen der Eingebetteten Systeme**

Studiengang:	[909]	Modulkürzel:	051711010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Martin Radetzki

Dozenten:	<ul style="list-style-type: none"><li>• Martin Radetzki</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 3, 5. Semester</li></ul>
Lernziele:	Methodische und werkzeugunterstützte Konstruktion von Hardware und Software für auf spezifische Anwendungen hin optimierte eingebettete Systeme.
Inhalt:	Modelle zur Systemspezifikation, Modellierungssprachen und ihre Simulation, System- und Architektursynthese, Allokation von Ressourcen und Bindung von Aufgaben/Operationen, Hardware-Software-Partitionierung, Verfahren zur Ablaufplanung für parallele Architekturen, Optimierungsverfahren, Anwendungsspezifische Prozessoren, On-Chip/Board-Verbindungsnetzwerke.
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• J. Teich, Digitale Hardware/Software-Systeme, 2. Auflage, 2007</li><li>• P. Marwedel, Embedded System Design, 2006</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 100901 Vorlesung Grundlagen der Eingebetteten Systeme</li><li>• 100902 Übung Grundlagen der Eingebetteten Systeme</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine.
Prüfungsleistungen:	Bewertete Rechnerübungen - vorlesungsbegleitend (0.25) Schriftliche Prüfung von 90 Minuten Dauer (0.75)



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 10091 Grundlagen der Eingebetteten Systeme
- 10092 Grundlagen der Eingebetteten Systeme -  
Rechnerübungen

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10100 Grundlagen der Graphischen Ingenieursysteme**

---

Studiengang:	[909]	Modulkürzel:	051400005
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Dieter Roller

---

---

Dozenten:	<ul style="list-style-type: none"><li>•</li><li>• Dieter Roller</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 4. Semester</li></ul>
Lernziele:	<ul style="list-style-type: none"><li>• Kenntnis und Verständnis von Modellen bei der Produktentwicklung</li><li>• Grundkenntnisse über die wichtigsten Modellarten, Algorithmen und Datenstrukturen und Techniken für den Datenaustausch</li></ul>
Inhalt:	<p>Inhalte:</p> <ul style="list-style-type: none"><li>• Anforderungen an CAD-Systeme</li><li>• zweidimensionale Modelle</li><li>• dreidimensionale Modelle</li><li>• interaktive Modellerstellung</li><li>• Einführung in die Modifikationstechnik u. parametrische Modellierung</li><li>• Methoden zur Modellmodifikation</li><li>• Grundlagen der parametrischen Modellierung</li><li>• Ansätze und Verfahren zur parametrischen Variantenerzeugung</li><li>• Ausgewählte Anwendungsbeispiele</li><li>• Überblick über weitergehende Modellieransätze</li><li>• Datenverwaltung in CAD</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• D. Roller, CAD - Effiziente Anpassungs- und Variantenkonstruktion, Springer-Verlag</li><li>• Literatur, siehe Webseite zur Veranstaltung</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101001 Vorlesung Grundlagen der Graphischen Ingenieursysteme</li><li>• 101002 Übung Grundlagen der Graphischen Ingenieursysteme</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

keine

Prüfungsleistungen:

Schriftliche Prüfung von 60 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10101 Grundlagen der Graphischen Ingenieursysteme

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10110 Grundlagen der Künstlichen Intelligenz**

Studiengang:	[909]	Modulkürzel:	051900205
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Gunther Heidemann

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester

Lernziele:

Der Student / die Studentin beherrscht die Grundlagen der Künstlichen Intelligenz, kann Probleme der KI selbständig einordnen und mit den erlernten Methoden und Algorithmen bearbeiten.

Inhalt:

- Intelligenz
- Agentenbegriff
- Problemlösen durch Suchen, Suchverfahren
- Probleme mit Rand- und Nebenbedingungen
- Spiele
- Aussagen- und Prädikatenlogik
- Logikbasierte Agenten, Wissensrepräsentation
- Inferenz
- Planen
- Unsicherheit, probabilistisches Schließen
- Probabilistisches Schließen über die Zeit
- Sprachverarbeitung
- Entscheidungstheorie
- Lernen

Literatur / Lernmaterialien:

- G. F. Luger, Künstliche Intelligenz, 2001
- S. Russell, P. Norvig, Künstliche Intelligenz, 2004

Lehrveranstaltungen und  
-formen:

- 101101 Vorlesung Grundlagen der Künstlichen Intelligenz
- 101102 Übung Grundlagen der Künstlichen Intelligenz

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden



## Modulhandbuch Bachelor of Science Softwaretechnik

Studienleistungen:

Prüfungsvorleistung: Übungsschein

Prüfungsleistungen:

Schriftliche Prüfung von 120 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10111 Grundlagen der Künstlichen Intelligenz

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung
- BA (Komb) Informatik

**Modul 10120 Grundlagen der Modellbildung und Simulation**

Studiengang:	[909]	Modulkürzel:	051240010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Peter Bastian

Dozenten: • Peter Bastian

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 4. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 4. Semester

Lernziele:

Beherrschung des grundsätzlichen Vorgehens in der Modellbildung, Kenntnis einer Auswahl diskreter und kontinuierlicher Modelle und deren Simulation, Beherrschung der Fehler- und Effizienzanalyse numerischer Verfahren.

Inhalt:

Diese Vorlesung bietet eine Einführung in die Grundlagen der Modellbildung und Simulation mit dem Ziel der Vorbereitung auf weiterführende Vorlesungen in diesem Bereich. Da Simulationsmethoden oft für viele verschiedene Problemklassen einsetzbar sind ist die Vorlesung methodisch strukturiert. Nach einer Einführung des Modellbegriffs wird auf die Grundlagen der ereignisgesteuerten Simulation sowie auf zelluläre Automaten eingegangen. Den Hauptteil der Vorlesung bilden dann kontinuierliche Modelle sowie deren numerische Behandlung. Am Beispiel der Analyse elektrischer Netzwerke werden lineare Gleichungssysteme (direkte Methoden, direkte Methoden für dünnbesetzte Matrizen sowie Krylovraumverfahren), differentiell-algebraische Gleichungen (Existenz von Lösungen und Stabilität, Einschrittverfahren, Schrittweitenkontrolle, steife Probleme) besprochen. Den Abschluss bildet eine kurze Einführung in Eigenschaften und Lösung partieller Differentialgleichungen (Typeinteilung, Finite Differenzen).

Literatur / Lernmaterialien:

- Peter Bastian, Grundlagen der Modellbildung und Simulation, 2008



## Modulhandbuch Bachelor of Science Softwaretechnik

Lehrveranstaltungen und  
-formen:

- 101201 Vorlesung Grundlagen der Modellbildung und Simulation
- 101202 Übung Grundlagen der Modellbildung und Simulation

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer; bei geringer  
Teilnehmerzahl mündliche Prüfung von 30 Minuten Dauer.

Prüfungsnummer/n und  
-name:

- 10121 Grundlagen der Modellbildung und Simulation

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10140 Grundlagen der Rechnerarchitektur**

---

Studiengang:	[909]	Modulkürzel:	051700010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Hans-Joachim Wunderlich

---

---

Dozenten:	<ul style="list-style-type: none"><li>• Hans-Joachim Wunderlich</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 5. Semester</li></ul>
Lernziele:	<ul style="list-style-type: none"><li>• Grundlegendes Verständnis der Herausforderungen bei Einsatz, Entwurf und Fertigung moderner Prozessoren</li><li>• Kenntnis über den Zusammenhang von Kosten, Performanz und Verlustleistung beim Entwurf von Prozessoren und der Konzeption komplexer Systeme</li><li>• Kenntnis und Bewertung aktueller Architekturkonzepte</li></ul>
Inhalt:	<p>Es werden die klassischen Themen der Rechnerarchitektur als Hardware/Software-Schnittstelle sowie weiterführende technologische Themen behandelt.</p> <p>Dazu gehören insbesondere:</p> <ul style="list-style-type: none"><li>• Technologische Grundlagen: Entwurststile und Fertigungstechnik, Ausbeute, Test und Zuverlässigkeit, Kosten und Qualität, Skalierung.</li><li>• Performanz: Taktfrequenz und Befehle pro Takt, Geschwindigkeitsanalyse und -optimierung.</li><li>• Verlustleistung: Verlustleistungsanalyse, Optimierung von Verlustleistung und Performanz, Verlustleistung und Skalierung.</li><li>• Computerarithmetik: Effiziente Hardwareimplementierung der Grundrechenarten, Hardwareimplementierung spezieller Funktionen wie Exponentialfunktion, Logarithmus und trigonometrische Funktionen, Gleitkommaoperationen und Standards, arithmetische Pipelines und Filter, aktuelle Gleitkommaeinheiten wie SPE des Cell Prozessors oder SPARC.</li><li>• Parallelität auf Befehlsebene: Superskalarrechner, statisches und dynamisches Scheduling, Out-of-Order Ausführung und VLIW-Rechner, Multithreading</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

---

	<ul style="list-style-type: none"><li>• Parallele Architekturen: Shared Memory und Message Passing, Multi-Core Prozessoren und Multi-Core Systeme auf einem Chip.</li><li>• Speicherhierarchie: Speichertechnologie und Cacheentwurf.</li><li>• Fehlertoleranztechniken: Einzelprozessoren und Mehrprozessorsysteme.</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• I. Koren, Computer Arithmetic Algorithms, 2001</li><li>• J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 2006</li><li>• S. Iman, M. Pedram, Logic Synthesis for Low Power VLSI Designs, 1997</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101401 Vorlesung Grundlagen der Rechnerarchitektur</li><li>• 101402 Übung Grundlagen der Rechnerarchitektur</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10141 Grundlagen der Rechnerarchitektur</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10150 Grundlagen des Compilerbaus und der Programmiersprachen**

Studiengang:	[909]	Modulkürzel:	051510015
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Erhard Plödereder

Dozenten: • Erhard Plödereder

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, 5. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, 5. Semester

Pflichtvorlesung für die Vertiefungslinie Programmiersprachen und Compilerbau.

Lernziele:

Die Studierenden haben die Grundkenntnisse erlangt, die zur effizienten Verwendung von Lexer- und Parser-Generatoren zur Analyse von Eingabetexten nötig sind. Sie haben gelernt, die Fehlermeldungen aus diesen Generatoren und den Compilern oder Interpretern richtig einzuordnen. Ferner haben sie durch Betrachtung der Implementierungsmodelle typischer Programmiersprachenkonstrukte Verständnis für das Ausführungsverhalten und für typische, gefährliche Fehlerquellen in Anwendungsprogrammen erlangt.

Inhalt:

Compilerarchitekturen im Überblick; lexikalische und syntaktische Analyse von Texten mit formaler Grammatik, insb. von Programmiersprachen. Lexikalische Analyse: endliche Automaten und ihre Implementierung; Syntaxanalyse: diverse Parser-Strategien, ihre Implementierung und Eigenschaften. Methoden der automatischen Generierung von Analysatoren aus Spezifikationen der Grammatiken. Fehlererkennung und -behandlung. Analyse der statischen Semantik: Grundbegriffe und elementare Methoden. Eigenschaften von Programmiersprachen; Realisierung der Laufzeitsemantik prozeduraler Programmiersprachen aus Benutzersicht, insbesondere Implementierungsmodelle der Speicherverwaltung und der Unterprogrammaufrufe. Vermeidung typischer Fehlerquellen und überraschender Probleme in Anwendungsprogrammen.



## Modulhandbuch Bachelor of Science Softwaretechnik

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Aho, Sethi, Ullman, Compilers - Principles, Techniques, and Tools, 1988</li><li>• Wilhelm, Maurer, Uebersetzerbau, 1997</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 101501 Vorlesung Grundlagen des Compilerbaus und der Programmiersprachen</li><li>• 101502 Übung Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Studienbegleitende Prüfung
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10151 Grundlagen des Compilerbaus und der Programmiersprachen</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Physik</li><li>• B.Sc. Softwaretechnik</li><li>• BA (Komb) Informatik</li></ul>

**Modul 10170 Imaging Science**

Studiengang:	[909]	Modulkürzel:	051900210
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Gunther Heidemann

Dozenten:	<ul style="list-style-type: none"><li>• Paul Levi</li><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 1, Katalog STG, 5. Semester</li></ul>
Lernziele:	Der Student / die Studentin beherrscht die Grundlagen der Repräsentation und Verarbeitung digitaler Bilder, kann Probleme aus dem Fachgebiet einordnen und selbständig mit den erlernten Algorithmen und Verfahren lösen.
Inhalt:	<ul style="list-style-type: none"><li>• Grundlagen aus der Optik: Lochkamera, Linsengleichung</li><li>• Bildaufnahme: Kameras, Objektive, Beleuchtung, Aufnahmeprozess</li><li>• Bildrepräsentation: Diskretisierung, Farbräume, Bildformate (z.B. ppm, gif, jpeg)</li><li>• Elementare Bildbearbeitung: Punktoperationen (z.B. Kontrastverstärkung, Binarisierung)</li><li>• Lineare und nichtlineare Filter: Faltung, morphologische Operatoren</li><li>• Fourierdarstellung: Fouriertransformation, Bilddarstellung und -bearbeitung im Fourierraum, Abtasttheorem</li><li>• Orthogonale Transformationen: Cosinus, Wavelets</li><li>• Kompression: Generische Verfahren (RLE, Entropie), spezielle Bildverfahren (z.B. jpeg)</li><li>• Video: Formate, Kompression (z.B. AVI, MPEG)</li><li>• Bildverbesserung und Restauration</li><li>• Elementare Segmentierungsverfahren: Histogramme, Farben, Konturen</li><li>• Elementare Mustererkennung</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Bässmann, Henning; Kreys, Jutta, Bildverarbeitung Ad Oculos, 2004</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

- Forsyth, David and Ponce, Jean, Computer Vision. A Modern Approach.: A Modern Approach Computer Vision. A Modern Approach, 2003
- Gonzalez, Rafael C.; Woods, Richard E.; Eddins, Steven L., Digital Image Processing, 2004
- Jähne, Bernd, Digitale Bildverarbeitung, 2005
- Klaus D. Tönnies, Grundlagen der Bildverarbeitung, 2005
- L. G. Shapiro, G. C. Stockman, Computer Vision, 2001

Lehrveranstaltungen und  
-formen:

- 101701 Vorlesung Imaging Science
- 101702 Übung Imaging Science

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein

Prüfungsleistungen:

Schriftliche Prüfung von 120 Minuten Dauer

Prüfungsnummer/n und  
-name:

- 10171 Imaging Science

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10180 Information Retrieval und Text Mining**

Studiengang:	[909]	Modulkürzel:	052401010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Hinrich Schütze

Dozenten:

- Helmut Schmid
- Hinrich Schütze

Verwendbarkeit / Zuordnung zum Curriculum: Maschinelle Sprachverarbeitung Bachelor, Kernmodul, Pflicht, 5. Semester

Lernziele: Die Studierenden haben ein grundlegendes Verständnis der Konzepte und Algorithmen des Information Retrieval und Text Mining entwickelt.

Inhalt:

- Textpräprozessierung
- invertierte Indexe
- IR-Modelle (z.B. Vektorraum-basiertes IR)
- Linkanalyse
- Clustering
- Frage-Antwort-Systeme
- Informationsextraktion
- korpusbasierter Erwerb von lexikalischem und Weltwissen

Literatur / Lernmaterialien:

- Chris Manning, Prabhakar Raghavan, Hinrich Schütze, Introduction to Information Retrieval, 2008 Cambridge University Press

Lehrveranstaltungen und -formen:

- 101801 Vorlesung Information Retrieval and Text Mining
- 101802 Übung Information Retrieval and Text Mining

Abschätzung  
Arbeitsaufwand: Präsenzzeit: 42 Stunden  
Selbststudium: 138 Stunden

Studienleistungen: regelmäßige Hausübungen

Prüfungsleistungen: Klausur, Gewicht 1,0, schriftlich, 60 Minuten



## Modulhandbuch Bachelor of Science Softwaretechnik

Prüfungsnummer/n und  
-name:

- 10181 Information Retrieval und Text Mining

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Maschinelle Sprachverarbeitung
- BA (Komb) Informatik

**Modul 10210 Mensch-Computer-Interaktion**

Studiengang:	[909]	Modulkürzel:	051900001
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Thomas Ertl

Dozenten:	<ul style="list-style-type: none"><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li><li>• Bachelor Softwaretechnik, Katalog INF 1, Katalog STG, Kernmodul, 4. Semester</li></ul>
Lernziele:	<p>Verständnis der Methoden und Konzepte der Mensch-Computer-Interaktion, insbesondere der graphisch-interaktiven Systeme:</p> <ul style="list-style-type: none"><li>• Architektur und technische Funktionsweise von GUI-Systemen</li><li>• kognitive Grundlagen und Konsequenzen für die Software-Ergonomie</li><li>• praktische Erfahrung mit der Erstellung von Benutzungsoberflächen mit verschiedenen Programmierschnittstellen</li></ul>
Inhalt:	<ul style="list-style-type: none"><li>• Einführung: Bezug zu anderen Gebieten, historische Entwicklung</li><li>• Menschliche Aspekte: sensorische (insbesondere) visuelle Wahrnehmung, Motorik, Gedächtnis-, Aufmerksamkeits- und Problemlösungsmodelle</li><li>• Computer-Aspekte: Ein/Ausgabegeräte, Display-Architekturen und Event-Verarbeitung, Multimedia-Grundlagen (Vektor-/Rastergraphik, Audio/Video, Farbsysteme), 2D-Graphik (Compositing, Rasterisierung, Linien, Polygone, Text, Bilder, APIs)</li><li>• Interaktionskonzepte und -stile: Geräte- vs. Task-Ebene, Kommandozeile, Menüs, Formulare, Gestik, Spracheingabe, graphische Stile: Direkte Manipulation, WYSIWYG, Icons</li><li>• Fenstersysteme und GUI Toolkits Basisaspekte (Fenstermanagement, Event-Zuordnung), Schichtenaufbau (X, WPF), Widgets/Componenten, Toolkit-Architektur (z.B. Qt, AWT/Swing, XML/Web-basiert), Verwendung von Standardkomponenten</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

- Software Ergonomie, Entwurfsprinzipien: Normen, Regeln (Shneidermann), Style Guides, Modelle (MVC), Metaphern, Entwicklungswerkzeuge
- Evaluation: Experten-Evaluation (Walkthrough, GOMS), Benutzer-Evaluation (Think-aloud, Interviews, Auswertung)
- Spezielle Systeme: mobile Geräte, Virtual/Augmented Reality

Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.

Literatur / Lernmaterialien:

- Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004
- Ben Shneiderman, Catherine Plaisant, Designing the User Interfacs, 2005
- Bernhard Preim, Entwicklung interaktiver Systeme, 1999

Lehrveranstaltungen und -formen:

- 102101 Vorlesung Mensch-Computer-Interaktion
- 102102 Übung Mensch-Computer-Interaktion

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Prüfungsvorleistung: Übungsschein.

Prüfungsleistungen:

Modulprüfung: Schriftliche Prüfung, 120 Minuten.

Prüfungsnummer/n und -name:

- 10211 Mensch-Computer-Interaktion

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Wirtschaftsinformatik
- B.Sc. Maschinelle Sprachverarbeitung

**Modul 10250 Parallele Systeme**

Studiengang:	[909]	Modulkürzel:	051200065
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Sven Simon

Dozenten:	<ul style="list-style-type: none"><li>• Sven Simon</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog INF 2, Katalog STG, 3. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog INF 2, Katalog STG, 3. Semester</li></ul>
Lernziele:	Grundlegende Kenntnisse im Bereich paralleler Systeme, z.B. Multi-Core CPUs und deren Programmierung.
Inhalt:	<ul style="list-style-type: none"><li>• Die Entwicklung vom klassischen Mikroprozessor zur Multi-Core CPU</li><li>• Programmierung paralleler Rechnersysteme</li><li>• Systolische Arrays, massiv parallele Systeme</li><li>• Parallele Systeme aus verschiedenen</li><li>• Anwendungsdomänen: ausgewählte Fallbeispiele</li></ul>
Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• Literatur, siehe Webseite zur Veranstaltung</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102501 Vorlesung Parallele Systeme</li><li>• 102502 Übung Parallele Systeme</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10251 Parallele Systeme</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10300 Rechnernetze**

Studiengang:	[909]	Modulkürzel:	051200010
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Kurt Rothermel

Dozenten: • Kurt Rothermel

Verwendbarkeit / Zuordnung zum Curriculum: • Bachelor Informatik, Ergänzungsmodul, Katalog Wahl-INF 3, 4. Semester  
• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog Wahl-INF 3, 4. Semester

Lernziele: • Versteht grundsätzliche Eigenschaften, Konzepte und Methoden von Rechnernetzen, insbesondere dem Internet.  
• Versteht Schichten und deren Zusammenwirken in einem Protokollstapel  
• Kann Rechnernetze aufbauen, verwalten und analysieren.  
• Kann Protokolle entwickeln und in Schichtenarchitektur einbetten.  
• Kann höhere Kommunikationsdienste zur Entwicklung von netzgestützten Systemen anwenden.  
• Kann sich mit Experten anderer Domänen über Methoden der Rechnernetze verständigen.

Inhalt: • Einführung in die Rechnernetze, ISO Referenzmodell;  
• Bitübertragungsschicht: Übertragungsmedien, analoge und digitale Informationskodierung und -übertragung, Vermittlungsarten;  
• Sicherungsschicht: Betriebsarten, Fehlererkennung und -behandlung, Flusskontrolle;  
• Lokale Netze: CSMA/CD, Token Ring, Token Bus, FDDI, Kopplung;  
• Vermittlungsschicht: Verbindungsorientierter und verbindungsloser Dienst, Leitwegbestimmung, Überlastkontrolle;  
• Internetworking;  
• Internet-Protokoll;  
• Transportschicht: ausgewählte Realisierungsprobleme und Internet-Protokolle;  
• Echtzeitkommunikation: IntServ, DiffServ; Sicherheit: Verfahren, IPsec, SSL, TLS.

Literatur / Lernmaterialien: • A.S. Tanenbaum, Computer Networks, 4th Edition, 2003  
• D.E. Comer, Computernetzwerke und Internets, 2000



## Modulhandbuch Bachelor of Science Softwaretechnik

- D.E. Comer, Internetworking with TCP/IP Volume I: Principles, Protocols, and Architecture, 1995
- J. F. Kurose, K. W. Ross, Computer Networks: a top-down approach featuring the Internet, 2001
- L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach, 1999

Lehrveranstaltungen und -formen:

- 103001 VL Rechnernetze
- 103002 ÜB Rechnernetze

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer (0.7)  
Benotete Übungen - studienbegleitend (0.3)

Prüfungsnummer/n und -name:

- 10301 Rechnernetze
- 10302 Rechnernetze - Übungen

Studiengänge die dieses Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 10330 Systemkonzepte und -programmierung**

Studiengang:	[909]	Modulkürzel:	051200005
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	Kurt Rothermel

Dozenten:

- Kurt Rothermel
- Frank Leymann

Verwendbarkeit / Zuordnung zum Curriculum:

- Bachelor Informatik, Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Kernmodul, 3. Semester

Lernziele:

- Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen
- Verstehen systemnaher Konzepte und Mechanismen
- Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.
- Kann systemnahe Software entwerfen und implementieren.
- Kann nebenläufige Programme entwickeln
- Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.

Inhalt:

Grundlegende Systemstrukturen - und organisationen

- zentrale, verteilte, parallele Systeme
- client/server, Producer/Consumer, P2P, Grid
- Betriebssysteme, Systemplattformen (Middleware), Kerne für eingebettete Systeme

Grundlagen der Rechnernetze

- Grundlegende Netzkonzepte und -architekturen
- Grundlegende Kommunikationsprotokolle und -dienste
- Prozessmanagement
- Prozessbeschreibung, -kontrolle
- Threads

Interprozesskommunikation

- Gemeinsamer Speicher
- Message Passing (Messages, RPC/RMI, Message Queuing, Ereignisse)



## Synchronisation

- Abstraktionen für Shared Memory (Semaphore, Monitore, ...)
- Abstraktionen für Message Passing
- Verklemmungen (Modelle, Behandlung)

## Kausalität und logische Uhren Scheduling

- Uniprozessor
- Multiprozessor
- Realtime

## Schutz und Sicherheit

- grundlegende Konzepte der Sicherheit
- ACLs, Capabilities

## Literatur / Lernmaterialien:

- Literatur, siehe Webseite zur Veranstaltung

## Lehrveranstaltungen und -formen:

- 103301 Vorlesung Systemkonzepte und -programmierung
- 103302 Übung Systemkonzepte und -programmierung

## Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

## Studienleistungen:

Keine

## Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer (Faktor 0.7)  
Benoteter Übungsschein (Faktor 0.3)

## Prüfungsnummer/n und -name:

- 10331 Systemkonzepte und -programmierung
- 10332 Systemkonzepte und -programmierung -  
Übungsschein

## Studiengänge die dieses Modul nutzen :

- B.Sc. Informatik
- B.Sc. Physik
- B.Sc. Softwaretechnik
- B.Sc. Technikpädagogik
- M.Sc. Technikpädagogik

**Modul 11330 Visualisierung**

Studiengang:	[909]	Modulkürzel:	051900011
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	Deutsch	Modulverantwortlicher:	Daniel Weiskopf

Dozenten:	<ul style="list-style-type: none"><li>• Thomas Ertl</li><li>• Daniel Weiskopf</li><li>• Carsten Dachsbacher</li></ul>
Verwendbarkeit / Zuordnung zum Curriculum:	<ul style="list-style-type: none"><li>• Bachelor Informatik, Ergänzungsmodul, Katalog Wahl-INF 1, 6. Semester</li><li>• Bachelor Softwaretechnik, Ergänzungsmodul, Katalog Wahl-INF 1, 6. Semester</li></ul>
Lernziele:	Die Studierenden haben Wissen über Grundlagen, Algorithmen und Datenstrukturen für die Visualisierung sowie praktische Fähigkeiten durch die Arbeit mit Visualisierungssoftware erworben.
Inhalt:	<p>Visualisierung behandelt alle Aspekte, die mit der visuellen Repräsentation von Daten aus wissenschaftlichen Experimenten, Simulationen, medizinischen Scannern, Datenbanken oder ähnlichen Datenquellen gewonnen werden, um zu einem tieferen Verständnis zu gelangen oder eine einfachere Darstellung komplexer Phänomene oder Sachverhalte zu erhalten. Um dieses Ziel zu erreichen, werden zum einen wohlbekannte Techniken aus dem Gebiet der interaktiven Computergraphik, zum anderen auch neu entwickelte Techniken angewendet.</p> <p>Entsprechend werden in dieser Vorlesung folgenden Themen behandelt:</p> <ul style="list-style-type: none"><li>• Einführung, Historie, Visualisierungspipeline</li><li>• Datenakquise und -repräsentation (Abtasten, Rekonstruktion, Gitter, Datenstrukturen)</li><li>• Wahrnehmungsaspekte</li><li>• Grundlegende Konzepte visueller Abbildungen</li><li>• Visualisierung von Skalarfeldern (Isoflächenextraktion, Volumenrendering)</li><li>• Visualisierung von Vektorfelder (Teilchenverfolgung, texturbasierte Methoden, Topologie)</li><li>• Tensorfelder, Multiattributdaten</li><li>• Hochdimensionale Daten und Informationsvisualisierung</li></ul>



## Modulhandbuch Bachelor of Science Softwaretechnik

---

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• C. D. Hansen, C. R. Johnson, The Visualization Handbook, 2005</li><li>• C. Ware, Information Visualization: Perception for Design, 2004</li><li>• H. Schumann, W. Müller, Visualisierung: Grundlagen und allgemeine Methoden, 2000</li><li>• K. Engel, M. Hadwiger, J. M. Kniss, C. Rezk-Salama, D. Weiskopf, Real-time Volume Graphics, 2006</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 113301 Vorlesung Visualisierung</li><li>• 113302 Übungen Visualisierung</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Prüfungsvorleistung: Übungsschein.
Prüfungsleistungen:	Mündliche Prüfung von 30 Minuten Dauer.
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 11331 Visualisierung</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Softwaretechnik</li><li>• B.Sc. Maschinelle Sprachverarbeitung</li><li>• BA (Komb) Informatik</li></ul>

**Modul 14380 Hardware Verification and Quality Assessment**

Studiengang:	[909]	Modulkürzel:	051700020
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog ISW 2 (Hinweis: PL, 6 LP), Katalog Wahl-INF 2 (Hinweis: PL, 6 LP), 4. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog ISW 2 (Hinweis: PL, 6 LP), Katalog Wahl-INF 2 (Hinweis: PL, 6 LP), 4. Semester
- optional subject in index for Master of INFOTECH

Lernziele:

- Basic knowledge of methodologies and algorithms of functional and formal verification, diagnosis, test and design for testability of integrated circuits
- Application of tools for simulation, verification and test insertion

Inhalt:

Complex integrated circuits and systems are hardly designed fault free at first go. Also during production defects and an imperfect yield have to be expected. The course deals with the basic techniques to find and locate faults and defects in the design and in the manufactured, integrated system. The discussed methods are applied with the help of commercial and academic tools in exercises and labs.

The course comprises:

- Validation: Simulation and emulation in different design levels.
- Formal verification: Equivalence checking and model checking.
- Test: Fault simulation and test generation.
- Debug and diagnosis.

Literatur / Lernmaterialien:

- G. D. Hachtel, F. Somenzi, Logic Synthesis and Verification Algorithms, 2006
- K. L. McMillan, Symbolic Model Checking, 1993
- L.-T. Wang, C.-W. Wu, X. Wen, VLSI Test Principles and Architectures - Design for Testability, 2006
- M. L. Bushnell, V. D. Agrawal, Essentials of Electronic Testing, 2005
- R. Drechsler, B. Becker, Graphenbasierte Funktionsdarstellung, 2000



## Modulhandbuch Bachelor of Science Softwaretechnik

- S. Hassoun, T. Sasao, Logic Synthesis and Verification, 2002
- S. Minato, Binary Decision Diagrams and Applications for VLSI CAD, 1996
- T. Kropf, Introduction to Formal Hardware Verification, 1999
- W. Kunz, D. Stoffel, Reasoning in Boolean Networks, 1997

Lehrveranstaltungen und  
-formen:

- 143801 Vorlesung Hardware Verification and Quality Assessment
- 143802 Übung Hardware Verification and Quality Assessment

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 42 Stunden  
Nachbearbeitungszeit: 138 Stunden

Studienleistungen:

Keine

Prüfungsleistungen:

Schriftliche Prüfung von 90 Minuten Dauer

Prüfungsnummer/n und  
-name:

- 14381 Hardware Verification and Quality Assessment

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Informatik
- B.Sc. Softwaretechnik
- BA (Komb) Informatik

**Modul 16790 Rechnerorganisation 1**

Studiengang:	[909]	Modulkürzel:	051700006
Leistungspunkte:	6.0	SWS:	4.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, SoSe
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Informatik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Bachelor Softwaretechnik, Ergänzungsmodul, Katalog SWT 3 (Hinweis: LBP, PL, 6 LP), Kernmodul, 3. Semester
- Der erste Teil des Moduls Rechnerorganisation im Studiengang Informatik
- Eigenständiges Modul im Studiengang Softwaretechnik

Lernziele:

- Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung
- Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen

Inhalt:

Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Experimente mit Prozessorsimulatoren vertieft. Im einzelnen werden behandelt:

- Grundstrukturen: Stack-, Akkumulator- und Register-basierende Rechner
- Informationsdarstellung in Hardware: Daten- und Befehlsformate, Fehlererkennung und -korrektur, Befehlssätze und Unterstützung von Hochsprachen.
- MIPS als RISC-Bespiel und seine Assemblerprogrammierung
- Grundelemente und Entwurf auf Register-Transfer-Ebene
- Grundzüge einer Hardware-Beschreibungssprache (VHDL)
- Operationswerke: Multiplikation, Division, Gleitkommaeinheiten
- Steuerwerksentwurf und Mikroprogrammierung
- Befehlszyklus und Unterbrechungen
- Pipelining und statisches Scheduling
- Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung, Segmentierung, TLB, MMU und DMA
- Leistungsbewertung: Maßzahlen und CPI, Benchmarking und einfache Anwendung von Warteschlangen



## Modulhandbuch Bachelor of Science Softwaretechnik

Literatur / Lernmaterialien:	<ul style="list-style-type: none"><li>• D. A. Patterson, J. L. Hennessy, Computer Organization: The Hardware-Software Interface, 2004</li><li>• J. P. Hayes, Computer Architecture and Organization, 1998</li></ul>
Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 167901 Vorlesung Rechnerorganisation 1</li><li>• 167902 Praktische und theoretische Übungen Rechnerorganisation 1</li></ul>
Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
Studienleistungen:	Keine
Prüfungsleistungen:	Übungsaufgaben, 0.30, Lehrveranstaltungsbegleitende Prüfung Klausurarbeit, 0.70, schriftlich, 60 Minuten
Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 16791 Rechnerorganisation 1</li><li>• 16792 Rechnerorganisation 1 - Übungsaufgaben</li></ul>
Studiengänge die dieses Modul nutzen :	<ul style="list-style-type: none"><li>• B.Sc. Informatik</li><li>• B.Sc. Softwaretechnik</li></ul>



**Modul 400 Schlüsselqualifikationen fachaffin**

zugeordnet zu: Studiengang

---

Zugeordnete Module:                      16610    Studienprojekt-Pr

---

---

**Modul 16610 Studienprojekt-Pr**

Studiengang:	[909]	Modulkürzel:	051520191
Leistungspunkte:	15.0	SWS:	10.0
Moduldauer:	1 Semester	Turnus:	unregelmäßig
Sprache:	-	Modulverantwortlicher:	

Dozenten:

Verwendbarkeit /  
Zuordnung zum  
Curriculum:

- Bachelor Softwaretechnik, Fachaffine SQ, Kernmodul, 4. Semester

Diese Veranstaltung ist Teil des Moduls "Studienprojekt". Die Teilnehmer arbeiten als Team am Projekt. Die Organisation der Arbeit, auch die Projektleitung, ist Sache der Teilnehmer. Sie werden von den Betreuern beraten und unterstützt.

Lernziele:

Im Studienprojekt-Pr werden die Prinzipien der Kooperation in einem größeren, für die Praxis typischen Projekt angewendet und eingeübt. Dazu gehören die Kontakte zum Kunden (Anforderungsanalyse), die Projektplanung, die Kostenschätzung, die Qualitätssicherung und die Präsentation der Resultate in schriftlicher und mündlicher Form, auch die Techniken zur Konfliktlösung und zum Risiko-Management. Natürlich kommt auch das fachliche Wissen zur Realisierung eines Softwaresystems zum Zuge. Die Teilnehmer sind nach dem Projekt in der Lage, ein größeres Softwareprojekt zu organisieren und vollständig durchzuführen.

Inhalt:

Die Teilnehmer entwickeln ein Softwaresystem nach Vorgaben des Kunden von der Angebotserstellung bis zur Übergabe. Störungen und Änderungen der Aufgabe im Projektverlauf sind normale Bestandteile des Projekts. Typisch beginnt das Studienprojekt mit der Erhebung der Anforderungen und der Anfertigung eines Angebots; darauf folgt die Entwicklung nach einem zu Beginn gewählten Prozessmodell. Das Projekt wird mit der Übergabe der Software in einer Präsentation abgeschlossen. Die Teilnehmer fertigen einen Bericht an, der die individuellen Leistungen erkennen lässt

Lehrveranstaltungen und  
-formen:

- 166101 Praktikum Studienprojekt-Pr



## Modulhandbuch Bachelor of Science Softwaretechnik

Abschätzung  
Arbeitsaufwand:

Präsenzzeit: 210 Stunden  
Nachbearbeitungszeit: 210 Stunden

Studienleistungen:

Ausreichende Leistungen im Studienprojekt-Pr werden in jedem der beiden Semester durch einen unbenoteten Schein bestätigt.

Prüfungsleistungen:

Die Leistungen im Studienprojekt-Pr werden auf der Grundlage der im Projekt gezeigten Leistungen und des Projektberichts, der die individuellen Beiträge der Teilnehmer angeben muss, bewertet.

Prüfungsnummer/n und  
-name:

- 16611 Studienprojekt-Pr

Studiengänge die dieses  
Modul nutzen :

- B.Sc. Softwaretechnik



**Modul 800 Schlüsselqualifikationen fachübergreifend gesamt**

zugeordnet zu: Studiengang

---

Zugeordnete Module:	900	Schlüsselqualifikationen fachübergreifend
	930340	English for Software Engineering

---

---

**Modul 900 Schlüsselqualifikationen fachübergreifend**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	2 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	Deutsch	Modulverantwortlicher:	

---

---

Zugeordnete Module	901	Kompetenzbereich 1: Methodische Kompetenzen
	902	Kompetenzbereich 2: Soziale Kompetenzen
	903	Kompetenzbereich 3: Kommunikative Kompetenzen
	904	Kompetenzbereich 4: Personale Kompetenzen
	905	Kompetenzbereich 5: Recht, Wirtschaft, Politik

Dozenten:



## **Modul 901 Kompetenzbereich 1: Methodische Kompetenzen**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Dozenten:



## **Modul 902 Kompetenzbereich 2: Soziale Kompetenzen**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Dozenten:



## **Modul 903 Kompetenzbereich 3: Kommunikative Kompetenzen**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Dozenten:



## **Modul 904 Kompetenzbereich 4: Personale Kompetenzen**

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Dozenten:



**Modul 905 Kompetenzbereich 5: Recht, Wirtschaft, Politik**

---

Studiengang:	[909]	Modulkürzel:	-
Leistungspunkte:	0.0	SWS:	0.0
Moduldauer:	1 Semester	Turnus:	jedes 2. Semester, WiSe
Sprache:	-	Modulverantwortlicher:	

---

---

Dozenten:



---

## **Modul 930340 English for Software Engineering**

Studiengang:	[SQ]	Modulkürzel:	9300034
Leistungspunkte:	3.0	SWS:	2.0
Moduldauer:	1 Semester	Turnus:	jedes Semester
Sprache:	Englisch	Modulverantwortlicher:	

---

Dozenten:

Lehrveranstaltungen und  
-formen:

- 930340 English for Software Engineering

Studiengänge die dieses  
Modul nutzen :

- ohne Absch Schlüsselqualifikation



**Modul 10160 Hardwarebeschreibungssprachen**

zugeordnet zu: Studiengang

---

---

---



**Modul 11900 Design and Test of Systems on a Chip**

zugeordnet zu: Studiengang

---

---

---