

Modulhandbuch
Studiengang Lehramt an Gymnasien (GymPO I) Informatik
Prüfungsordnung: 2010
Hauptfach

Wintersemester 2016/17
Stand: 11. Oktober 2016

Universität Stuttgart
Keplerstr. 7
70174 Stuttgart

Kontaktpersonen:

Studiendekan/in:	Univ.-Prof. Stefan Wagner Institut für Softwaretechnologie Tel.: 0711/685-88455 E-Mail: stefan.wagner@informatik.uni-stuttgart.de
Studiengangsmanager/in:	Katrin Schneider Institut für Maschinelle Sprachverarbeitung Tel.: 685 88520 E-Mail: katrin.schneider@informatik.uni-stuttgart.de
Prüfungsausschussvorsitzende/r:	Univ.-Prof. Stefan Funke Institut für Formale Methoden der Informatik Tel.: E-Mail: Stefan.Funke@informatik.uni-stuttgart.de
Fachstudienberater/in:	Univ.-Prof. Stefan Funke Institut für Formale Methoden der Informatik Tel.: E-Mail: Stefan.Funke@informatik.uni-stuttgart.de
Stundenplanverantwortliche/r:	Apl. Prof. Ulrich Hertrampf Institut für Formale Methoden der Informatik Tel.: 7816-344 E-Mail: ulrich.hertrampf@f05.uni-stuttgart.de

Inhaltsverzeichnis

Präambel	4
200 Pflichtmodule	5
27620 Algorithmen und Berechenbarkeit für Lehramt	6
12060 Datenstrukturen und Algorithmen	7
14360 Einführung in die Technische Informatik	9
10190 Mathematik für Informatiker und Softwaretechniker	11
27630 Praktische Informatik für Lehramt	13
10280 Programmierung und Software-Entwicklung	14
10320 Seminar-INF 1	16
27640 Systemkonzepte und -programmierung für Lehramt	18
10940 Theoretische Grundlagen der Informatik	19
300 Wahlmodule	21
10030 Architektur von Anwendungssystemen	22
10060 Computergraphik	24
10080 Datenbanken und Informationssysteme	26
10100 Grundlagen der Graphischen Ingenieursysteme	28
56930 Grundlagen der Rechnerarchitektur	29
25610 Grundlagen des Software Engineerings	31
42410 Grundlagen des Wissenschaftlichen Rechnens	33
10210 Mensch-Computer-Interaktion	34
10220 Modellierung	36
400 Fachdidaktikmodule	38
34050 Gestaltung von Lehr- / Lernprozessen im Informatik-Unterricht, Projekt	39
34060 Grundlagen der Fachdidaktik Informatik	40
3000 Zwischenprüfung	42
12060 Datenstrukturen und Algorithmen	43
10280 Programmierung und Software-Entwicklung	45
10940 Theoretische Grundlagen der Informatik	47

Präambel

Informatik ist die Wissenschaft von der Informationsverarbeitung und den informationsverarbeitenden Systemen. Sie umfasst deren Theorie und Methodik, den Einsatz dieser Systeme, aber auch die Auswirkungen. Die Informatik ist damit ein Grundpfeiler der modernen Informationsgesellschaft. Informatiksysteme durchdringen unser tägliches Leben. Was noch vor wenigen Jahren unvorstellbar war, ist heute selbstverständlicher Standard. Die weltweite freie Bereitstellung von Wissen und die Möglichkeit, sich ohne Kosten per E-Mail auszutauschen sowie riesige Datenmengen, etwa in Form von Musik und Filmen zu speichern, bedeutet eine gesellschaftliche Neuerung, an deren Gestaltung man durch ein Informatikstudium aktiv mitwirken kann.

Durch Verfahren der Modellbildung und Abstraktion formuliert die Informatik allgemeine Gesetze, die der Informationsverarbeitung zugrunde liegen, und sucht Standardlösungen für praxisrelevante Aufgaben. Von wachsender Bedeutung wird dabei die Beherrschung immer komplexer werdender verteilter und vernetzter Systeme. Informatikerinnen und Informatiker operieren mit abstrakten Zeichen und Objekten, untersuchen Daten-, Sprach- und Systemstrukturen und entwickeln formale Programmiersprachen zur Formulierung von Algorithmen, Prozessen, Systemen und speziellen Anwendungen. Die Hard- und Software-Systeme stehen dabei als Forschungsobjekte und gleichzeitig als Werkzeuge im Mittelpunkt der Arbeit. Durch Visualisierung und Simulation werden neue Anwendungen erschlossen. Informatik ist einerseits eine Strukturwissenschaft, andererseits dominieren aber heute die ingenieurwissenschaftlichen Methoden und Verfahren. Die Informatik an der Universität Stuttgart ist geprägt durch hohen Praxisbezug und Anwendungen, ohne dabei die notwendigen Grundlagen zu vernachlässigen.

200 Pflichtmodule

Zugeordnete Module:	10190	Mathematik für Informatiker und Softwaretechniker
	10280	Programmierung und Software-Entwicklung
	10320	Seminar-INF 1
	10940	Theoretische Grundlagen der Informatik
	12060	Datenstrukturen und Algorithmen
	14360	Einführung in die Technische Informatik
	27620	Algorithmen und Berechenbarkeit für Lehramt
	27630	Praktische Informatik für Lehramt
	27640	Systemkonzepte und -programmierung für Lehramt

Modul: 27620 Algorithmen und Berechenbarkeit für Lehramt

2. Modulkürzel:	050420021	5. Moduldauer:	1 Semester
3. Leistungspunkte:	7.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	• Stefan Funke • Volker Diekert • Ulrich Hertrampf		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Vorlesungen aus dem 1. und 2. Semester		
12. Lernziele:	Die Studierenden kennen die Klassifizierung von Algorithmen in effizient berechenbar, NP-vollständig, PSPACE-Algorithmen und prinzipielle Unberechenbarkeit. Sie haben wichtige Entwurfsstrategien und Analysemethoden kennengelernt.		
13. Inhalt:	Berechenbarkeit vs. Unberechenbarkeit, Church'sche These, NP-Vollständigkeit, PSPACE-vollständige Algorithmen (QBF). Entwurfsstrategien: Teile und Herrsche, gierig (greedy), Dynamisches Programmieren, Randomisierte Algorithmen		
14. Literatur:	• 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		
15. Lehrveranstaltungen und -formen:	276201 Vorlesung mit Übungen Algorithmen und Berechenbarkeit		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 h Selbststudiums- / Nacharbeitszeit: 168 h Summe: 210 h		
17. Prüfungsnummer/n und -name:	27621 Algorithmen und Berechenbarkeit für Lehramt (LBP), schriftlich, eventuell mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Formale Methoden der Informatik		

Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Daniel Weiskopf		
9. Dozenten:	• Andrés Bruhn • Thomas Ertl • Stefan Funke • Daniel Weiskopf		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10280 Programmierung und Software-Entwicklung		
12. 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. Die Lernziele lassen sich wie folgt zusammenfassen: • 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		
13. Inhalt:	Es werden die folgenden Themen behandelt: • Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen • Komplexität und Effizienz von Algorithmen, O-Notation • Listen (Stack, Queue, doppelt verkettete Listen) • Sortierverfahren (Selection-, Insertion-, Bubble-, Merge-, Quick-Sort) • Bäume (Binär-, AVL-, 2-3-4-, Rot-Schwarz-, B-Bäume, Suchbäume, Traversierung, Heap) • Räumliche Datenstrukturen (uniforme Gitter, Oktal-, BSP-, kD-, CSG-Bäume, Bounding-Volumes) • Graphen (Datenstrukturen, DFS, BFS, topologische Traversierung, Dijkstra-, A*-, Bellman-Ford-Algorithmen, minimale Spannbäume, maximaler Fluss) • Räumliche Graphen (Triangulierung, Voronoi, Delaunay, Graph-Layout) • Textalgorithmen (String-Matching, Knuth-Morris-Pratt, Boyer-Moore, reguläre Ausdrücke, Levenshtein-Distanz) • Hashing (Hashfunktionen, Kollisionen) • Verteilte Algorithmen (Petri-Netze, Programmieren nebenläufiger Abläufe, einige parallele und parallelisierte Algorithmen)		

	• Algorithmenentwurf und -muster (inkrementell, greedy, divide-and-conquer, dynamische Programmierung, Backtracking, randomisierte Algorithmen) • Maschinelles Lernen (überwachtes Lernen, Entscheidungsbäume, SVM, neuronale Netze; unüberwachtes Lernen, k-Means)								
14. Literatur:	• G. Saake, K. Sattler. <i>Algorithmen und Datenstrukturen: Eine Einführung mit Java</i> . 5. Auflage, dpunkt-Verlag, 2013 • T. Ottmann, P. Widmayer. <i>Algorithmen und Datenstrukturen</i> . 5. Auflage, Springer-Verlag, 2012								
15. Lehrveranstaltungen und -formen:	• 120601 Vorlesung Datenstrukturen und Algorithmen • 120602 Übung Datenstrukturen und Algorithmen								
16. Abschätzung Arbeitsaufwand:	<table> <tr> <td>Präsenzzeit:</td><td>63 h</td></tr> <tr> <td>Selbststudiums- /</td><td>207</td></tr> <tr> <td>Nachbearbeitungszeit:</td><td></td></tr> <tr> <td>Summe:</td><td>270 h</td></tr> </table>	Präsenzzeit:	63 h	Selbststudiums- /	207	Nachbearbeitungszeit:		Summe:	270 h
Präsenzzeit:	63 h								
Selbststudiums- /	207								
Nachbearbeitungszeit:									
Summe:	270 h								
17. Prüfungsnummer/n und -name:	• 12061 Datenstrukturen und Algorithmen (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben. • V Vorleistung (USL-V), schriftlich, eventuell mündlich								
18. Grundlage für ... :									
19. Medienform:									
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme								

Modul: 14360 Einführung in die Technische Informatik

2. Modulkürzel:	051400105	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Otto Eggenberger		
9. Dozenten:	• Otto Eggenberger • Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	Der/die Studierende kennt die grundlegende Funktionsweise eines Computers, versteht die elektrotechnischen Grundlagen und Technologien und kann einfache digitale Schaltungen analysieren, entwerfen und optimieren.		
13. Inhalt:	Grundlegende Funktionsweise eines Computers • Informationsdarstellung • Zahlendarstellung und Codes • Digitale Grundbausteine • Logische Funktionen, Speicherelemente • Befehlsausführung, Programmablauf Elektrotechnische Grundlagen • Physikalische Grundbegriffe • Elektrische Spannung, elektrischer Strom • Elektrische Netzwerke • Halbleiterbauelemente • Digitale Grundsaltungen Digitale Schaltungen • Schaltnetzwerke • Boolesche Algebra und Schaltalgebra • Darstellung und Minimierung von Schaltfunktionen • Rückkopplung, Zustandsbegriff • Automaten und sequentielle Netzwerke • Digitale Standardschaltungen • Entwurfsmethodik		
14. Literatur:	• Dirk W. Hoffman: Grundlagen der technischen Informatik, Hanser, 2007 • Bernd Becker, Rolf Drechsler, Paul Molitor: Technische Informatik, Pearson Studium, 2005 • Jörg Keller, Wolfgang J. Paul: Hardware Design, Teubner, 3. Aufl. 2005		
15. Lehrveranstaltungen und -formen:	• 143601 Vorlesung Einführung in die Technische Informatik • 143602 Gruppenübungen Einführung in die Technische Informatik		

16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h
	Selbststudiums- /	138 h
	Nachbearbeitungszeit:	
	Summe:	180 h
17. Prüfungsnummer/n und -name:	14361 Einführung in die Technische Informatik (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0	
18. Grundlage für ... :	36530 Rechnerorganisation 1	
19. Medienform:		
20. Angeboten von:	Institut für Rechnergestützte Ingenieursysteme	

Modul: 10190 Mathematik für Informatiker und Softwaretechniker

2. Modulkürzel:	080300100	5. Moduldauer:	2 Semester
3. Leistungspunkte:	18.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	12.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	PD Andreas Markus Kollross		
9. Dozenten:	• Peter Lesky • Wolfgang Rump • Wolf-Patrick Düll • Andreas Markus Kollross		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine, die Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Studierenden haben die mathematischen Grundlagen für die Studiengänge Informatik bzw. Softwaretechnik erarbeitet und den selbständigen und kreativen Umgang mit den mathematischen Stoffgebieten gelernt.		
13. Inhalt:	1. Semester: • 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)		
14. Literatur:	• Anna Sändig, Mathematik, Vorlesungsskripte , SS 2007 • D. Hachenberger, Mathematik für Informatiker, 2005 • M. Brill, Mathematik für Informatiker, 2001 • P.Hartmann, Mathematik für Informatiker, 2002		
15. 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		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 126 Stunden Nachbearbeitungszeit: 414 Stunden		
17. Prüfungsnummer/n und -name:	• 10191 Mathematik für Informatiker und Softwaretechniker (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Ein Übungsschein aus den beiden Veranstaltungen, jeweils im 1. oder 2. Fachsemester zu erwerben		

- V Vorleistung (USL-V), schriftlich, eventuell mündlich
-

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 27630 Praktische Informatik für Lehramt

2. Modulkürzel:	050420031	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Stefan Funke		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Lernziele der zugehörigen Veranstaltungen		
13. Inhalt:	Inhalte der zugehörigen Veranstaltungen		
14. Literatur:	wird in den Veranstaltungen angegeben		
15. Lehrveranstaltungen und -formen:	• 276301 Vorlesung Praktische Informatik für Lehramt (Teil A) • 276302 Vorlesung mit Übungen Modellierung (Teil B1) • 276303 Vorlesung mit Übungen Mensch-Computer-Interaktion (Teil B2)		
16. Abschätzung Arbeitsaufwand:	Summe: 360 h		
17. Prüfungsnummer/n und -name:	• 27631 Praktische Informatik für Lehramt B1 (PL), schriftlich, eventuell mündlich, Gewichtung: 1.0 • 27632 Praktische Informatik für Lehramt B2 (PL), schriftlich, eventuell mündlich, Gewichtung: 1.0 • 27633 Praktische Informatik für Lehramt, Projektschein (USL), schriftlich, eventuell mündlich, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Formale Methoden der Informatik		

Modul: 10280 Programmierung und Software-Entwicklung

2. Modulkürzel:	051520005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine. Teilnahme an einem Vorkurs Java ist hilfreich aber nicht notwendig.		
12. Lernziele:	Die Teilnehmer haben einen Überblick über das Gebiet der Informatik. Sie haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	• Die Programmiersprache Java und die virtuelle Maschine • Objekte, Klassen, Schnittstellen, Blöcke, Programmstrukturen, Kontrakte • Klassenmodellierung mit der UML • Objekterzeugung und -ausführung • Boolesche Logik • Verzweigungen, Schleifen, Routinen, Abstraktionen, Modularisierung, Variablen, Zuweisungen • Rechner, Hardware • Syntaxdarstellungen • Übersicht über Programmiersprachen und -werkzeuge • Grundlegende Datenstrukturen und Algorithmen • Vererbung, Polymorphe • Semantik • Programmierung graphischer Oberflächen • Übergang zum Software Engineering		
14. Literatur:	• Appelrath, Hans-Jürgen und Ludewig, Jochen, "Skriptum Informatik - eine konventionelle Einführung", Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999 • Meyer, Bertrand, "Touch of Class", Springer-Verlag, 2009 • Savitch, Walter, "Java. An Introduction to Problem Solving and Programming", Pearson, 6. Auflage, 2012		
15. Lehrveranstaltungen und -formen:	• 102801 Vorlesung Programmierung und Softwareentwicklung • 102802 Übung Programmierung und Softwareentwicklung		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	63 h	
	Selbststudiums- /	187 h	
	Nachbearbeitungszeit:		
	Prüfungsvorbereitung:	20 h	
	Summe:	270 h	

17. Prüfungsnummer/n und -name:	• 10281 Programmierung und Software-Entwicklung (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, • V Vorleistung (USL-V), schriftlich, eventuell mündlich, Vorleistung: Übungsschein. Voraussetzungen werden zu Beginn vom Dozenten festgesetzt. Dazu gehören eine bestimmte Anzahl von Vorträgen in den Übungen und ein bestimmter Teil der Übungspunkte.
18. Grundlage für ... :	12060 Datenstrukturen und Algorithmen
19. Medienform:	• Folien über Beamer • Tafelanschrieb
20. Angeboten von:	Software-Engineering

Modul: 10320 Seminar-INF 1

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

Ausarbeitung, sowie die aktive Mitarbeit während der Seminarveranstaltung.

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Softwaretechnologie

Modul: 27640 Systemkonzepte und -programmierung für Lehramt

2. Modulkürzel:	051200095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Univ.-Prof. Kurt Rothermel	
9. Dozenten:		• Kurt Rothermel • Frank Dürr	
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:			
12. Lernziele:			
13. Inhalt:			
14. Literatur:			
15. Lehrveranstaltungen und -formen:	276401	Vorlesung mit Übungen Systemkonzepte und -programmierung	
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	27641	Systemkonzepte und -programmierung für Lehramt (LBP), schriftliche Prüfung, 120 Min., Gewichtung: 1.0	
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 10940 Theoretische Grundlagen der Informatik

2. Modulkürzel:	050420005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	• Volker Diekert • Ulrich Hertrampf		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine		
12. 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.		
13. Inhalt:	Logik und Diskrete Strukturen: • Einführung in die Aussagenlogik: Semantik (Wahrheitswerte), Syntax (Axiome und Schlussregeln), Normalformen; Hornformeln; Endlichkeitssatz; aussagenlogische Resolution; • Einführung in die Prädikatenlogik 1. Stufe: Semantik und Syntax, Normalformen, Unifikatoren, Herbrand-Theorie, prädikatenlogische Resolution; • Elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, Euklidischer Algorithmus, Chinesischer Restsatz, Primzahltests, RSA-Verfahren; Wachstumsabschätzungen; Grundbegriffe der Wahrscheinlichkeitsrechnung; Kombinatorik; Graphen. Automaten und Formale Sprachen: • Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, Iterationslemmata für reguläre und kontextfreie Sprachen, Normalformen, Kellerautomaten, Lösen des Wortproblems kontextfreier Sprachen mit dem CYK-Algorithmus, linear beschränkte Automaten, kontextsensitive Grammatiken, Typ 0-Grammatiken und Turingmaschinen.		
14. Literatur:	• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988 • Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999		
15. 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	
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	84 h
	Selbststudiums- /	276 h
	Nachbearbeitungszeit:	
	Summe:	360 h
17. Prüfungsnummer/n und -name:	• 10941 Theoretische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein • V Vorleistung (USL-V), schriftlich, eventuell mündlich, 30 Min.	
18. Grundlage für ... :		
19. Medienform:		
20. Angeboten von:	Institut für Formale Methoden der Informatik	

300 Wahlmodule

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10060	Computergraphik
	10080	Datenbanken und Informationssysteme
	10100	Grundlagen der Graphischen Ingenieursysteme
	10210	Mensch-Computer-Interaktion
	10220	Modellierung
	25610	Grundlagen des Software Engineerings
	42410	Grundlagen des Wissenschaftlichen Rechnens
	56930	Grundlagen der Rechnerarchitektur

Modul: 10030 Architektur von Anwendungssystemen

2. Modulkürzel:	052010002	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	Die Vorlesung erläutert den Begriff der Architektur von Anwendungssystemen und die Rolle des Architekten solcher Systeme. Die wesentlichen Bestandteile von Anwendungsarchitektur wie etwa Datenbanksysteme, Anwendungsserver, Messaging Systeme, Workflowsysteme und TP-Monitore werden diskutiert. Die wesentlichen Mustern zur Erstellung von Anwendungssystemen sind verstanden.		
13. Inhalt:	Architekturelle Stile wie etwa N-stufige Aufbauten oder Service-Orientierung werden vorgestellt. Architekturmuster werden detailliert. Fundamentale Konzepte wie Transaktionen und Queuing werden eingeführt. Darauf aufbauend wird Direct TP vs Queues TP diskutiert. Grundlegende Qualitätseigenschaften wie Verfügbarkeit und Skalierbarkeit werden erläutert und Mechanismen zu deren Erzielen eingeführt. Die Rolle von Komponenten und Programmierung im Großen wird heraus gearbeitet und Modell-getriebene Architektur vorgestellt.		
14. Literatur:	• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002 • B. Neubauer, T. Ritter, F. Stoinnski, CORBA Komponenten, 2004 • F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, Pattern-orientierte Software Architektur - Ein Patternsystem, 1998 • F. Leymann, D. Roller, Production Workflow, 2000 • L. Hohmann, Beyond Software Architecture, 2003 • M. Fowler, Patterns of Enterprise Application Architecture, 2003 • P. Bernstein, E. Newcomer, Principles of Transaction Processing, 1997 • S. Conrad, W. Hasselbring, A. Koschel, R. Tritsch, Enterprise Application Integration, 2006 • S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005 • W. Emmerich, Konstruktion von verteilten Objekten, 2003		
15. Lehrveranstaltungen und -formen:	• 100301 Vorlesung Grundlagen der Architektur von Anwendungssystemen • 100302 Übung Grundlagen der Architektur von Anwendungssystemen		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		

17. Prüfungsnummer/n und -name:	10031 Architektur von Anwendungssystemen (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0,
18. Grundlage für ... :	• 29530 Business Process Management • 29480 Loose Coupling and Message Based Applications • 29510 Service Computing
19. Medienform:	Vorlesungen mit begleitenden Übungen
20. Angeboten von:	Architektur von Anwendungssystemen

Modul: 10060 Computergraphik

2. Modulkürzel:	051900002	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Thomas Ertl		
9. Dozenten:	• Thomas Ertl • Daniel Weiskopf • Martin Fuchs • Guido Reina		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10210 Mensch-Computer-Interaktion • Modul 41590 Einführung in die Numerik und Stochastik		
12. Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.		
13. Inhalt:	Folgende Themen werden in der Vorlesung behandelt: • Überblick über den Prozess der Bildsynthese • Graphische Geräte, visuelle Wahrnehmung, Farbsysteme • Grundlegende Rastergraphik und Bildverarbeitung • Raytracing und Beleuchtungsmodelle • 2D und 3D Geometrietransformationen, 3D Projektion • Graphikprogrammierung in OpenGL 3 • Texturen • Polygonale und hierarchische Modelle • Rasterisierung und Verdeckungsberechnung • Grundlagen der geometrischen Modellierung (Kurven, Flächen) • Räumliche Datenstrukturen Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.		
14. Literatur:	• J. Encarnacao, W. Strasser, R. Klein: Graphische Datenverarbeitung (Band1 und 2), 1997 • J. Foley, A. van Dam, S. Feiner, J. Hughes: Computer Graphics: Principle and Practice, 1990		
15. Lehrveranstaltungen und -formen:	• 100601 Vorlesung Computergraphik • 100602 Übung Computergraphik		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiums- /	138 h	
	Nachbearbeitungszeit:		
	Summe:	180 h	
17. Prüfungsnummer/n und -name:	• 10061 Computergraphik (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein. • V Vorleistung (USL-V), schriftlich, eventuell mündlich		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von: Institut für Visualisierung und Interaktive Systeme

Modul: 10080 Datenbanken und Informationssysteme

2. Modulkürzel:	051200025	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Bernhard Mitschang		
9. Dozenten:	• Bernhard Mitschang • Holger Schwarz		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Vorlesung Modellierung oder Gleichwertiges		
12. Lernziele:	Die Studierenden haben die erforderlichen Kenntnisse für Datenbankprogrammierer in angemessenem Umfang erworben.		
13. Inhalt:	Die Vorlesung "Datenbanken und Informationssysteme" ist als Einstiegsveranstaltung in das Vertiefungsgebiet Datenbanksysteme konzipiert. Aufbauend auf dem Inhalt der Vorlesung "Modellierung" werden insbesondere Entwurfs- und Realisierungsaspekte von Datenbanksystemen betrachtet. Die Entwicklung, Installation und Administration von Datenbanksystemen bestimmen hier sowohl Stoffauswahl als auch Detaillierungsgrad. Als Grundlage für alle weiteren Betrachtungen wird ein Schichtenmodell zur Beschreibung eines allgemeinen Datenbanksystems vorgestellt. Darauf aufbauend werden die einzelnen Systemschichten im Detail diskutiert, die dort zu realisierenden Komponenten betrachtet sowie die jeweils vorherrschenden Algorithmen beschrieben und bewertet. Im Einzelnen werden folgende Aspekte vertieft: Anwendungsprogrammierschnittstelle, Externspeicherverwaltung, DBS-Pufferverwaltung, Speicherungsstrukturen und Zugriffspfadstrukturen, Anfrageverarbeitung und Anfrageoptimierung, Transaktionsverarbeitung, Synchronisation, Logging und Recovery.		
14. Literatur:	• A. Kemper, A. Eickler, Datenbanksysteme - Eine Einführung, 2004 • Th. Härder, E. Rahm, Datenbanksysteme, 2008 • H. Garcia-Molina, J. D. Ullman, J. Widom, Database Systems. The Complete Book, 2003 • R. Elmasri, S. Navathe, Fundamentals of Database Systems, 2003		
15. Lehrveranstaltungen und -formen:	• 100801 Vorlesung Datenbanken und Informationssysteme • 100802 Übung Datenbanken und Informationssysteme		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 Stunden	
	Selbststudium:	138 Stunden	
17. Prüfungsnummer/n und -name:	10081	Datenbanken und Informationssysteme (PL), schriftlich oder mündlich, 60 Min., Gewichtung: 1.0, Prüfungsvorleistung: Modalitäten werden in der ersten Vorlesung angegeben	
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Modul: 10100 Grundlagen der Graphischen Ingenieursysteme

2. Modulkürzel:	051400005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dieter Roller		
9. Dozenten:	Dieter Roller		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Grundstudium		
12. Lernziele:	• Kenntnis und Verständnis von Modellen bei der Produktentwicklung • Grundkenntnisse über die wichtigsten Modellarten, Algorithmen und Datenstrukturen und Techniken für den Datenaustausch		
13. Inhalt:	Inhalte: • Anforderungen an CAD-Systeme • zweidimensionale Modelle • dreidimensionale Modelle • interaktive Modellerstellung • Einführung in die Modifikationstechnik u. parametrische Modellierung • Methoden zur Modellmodifikation • Grundlagen der parametrischen Modellierung • Ansätze und Verfahren zur parametrischen Variantenerzeugung • Ausgewählte Anwendungsbeispiele • Überblick über weitergehende Modellieransätze • Datenverwaltung in CAD		
14. Literatur:	• D. Roller, CAD - Effiziente Anpassungs- und Variantenkonstruktion, Springer-Verlag • Literatur, siehe Webseite zur Veranstaltung		
15. Lehrveranstaltungen und -formen:	• 101001 Vorlesung Grundlagen der Graphischen Ingenieursysteme • 101002 Übung Grundlagen der Graphischen Ingenieursysteme		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiums- /	138 h	
	Nachbearbeitungszeit:		
	Summe:	180 h	
17. Prüfungsnummer/n und -name:	10101 Grundlagen der Graphischen Ingenieursysteme (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Rechnergestützte Ingenieursysteme		

Modul: 56930 Grundlagen der Rechnerarchitektur

2. Modulkürzel:	-	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	• Michael Kochte • Hans-Joachim Wunderlich		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 41930 Rechnerorganisation		
12. Lernziele:	• Tiefes Verständnis von Entwurfskonzepten, die in modernen Prozessoren und Rechensystemen Verwendung finden • Kenntnis von Entwurfsherausforderungen. • Verständnis von aktuellen und zukünftigen Entwicklungstrends		
13. Inhalt:	Grundlegende und fortgeschrittene Themen der Rechenarchitektur, inklusive: • Technologiegrundlagen: Entwurfsverfahren, Herstellungsmethoden, Ausbeute, Test und Zuverlässigkeit, Kosten und Qualität, Skalierung. • Rechen- und Verlustleistung: Analyse und Optimierung • Rechnerarithmetik: Effiziente Hardwarestrukturen für grundlegende Arithmetik, Implementierung von Logarithmen, Exponentialfunktion und trigonometrischen Funktionen, arithmetische Pipelines, praktische Implementierungen von Gleitkommaarithmetik (Cell SPE, SPARC). • Instruktionsparallelismus (ILP): Superskalarität, statisches und dynamisches Scheduling, out-of-order execution, VLIW Prozessoren, Multithreading. • Datenparallelismus (DLP): Vektorprozessoren, SIMD, Grafikprozessoreinheiten (GPGPU) • Threadparallelismus (TLP): Mehrprozessorsysteme, Speicherkohärenz und Synchronisierung. • Speicher- und Cache-Architekturen: Entwurf und Optimierung		
14. Literatur:	• J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 2012 • I. Koren, Computer Arithmetic Algorithms, 2001 • Powerpoint Foliensatz • Auswahl von wissenschaftlichen Artikeln		
15. Lehrveranstaltungen und -formen:	• 569301 Vorlesung Grundlagen der Rechnerarchitektur • 569302 Übung Grundlagen der Rechnerarchitektur		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiums- /	138 h	
	Nachbearbeitungszeit:		
	Summe:	180 h	
17. Prüfungsnummer/n und -name:	56931 Grundlagen der Rechnerarchitektur (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von: Rechnerarchitektur

Modul: 25610 Grundlagen des Software Engineerings

2. Modulkürzel:	51520170	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Stefan Wagner		
9. Dozenten:	Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10280 Programmierung und Software-Entwicklung • Modul 12060 Datenstrukturen und Algorithmen • Programmiererfahrung		
12. Lernziele:	Die Teilnehmer kennen die Grundbegriffe des Software Engineerings und haben einen Überblick über die Methoden und Techniken, die dort angewandt werden. Einige ausgewählte Methoden und Techniken können angewandt werden.		
13. Inhalt:	Software Engineering kann in einer Vorlesung nicht erschöpfend behandelt werden. GSE gibt einen Überblick über das Gebiet und vertieft einzelne Themen, damit diese in der Praxis verwendet werden können. Es bildet damit auch die Basis für weitere Vertiefungen in diesem Gebiet. Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung. Die einzelnen Themen sind: • Geschichte und Konzepte des Software Engineerings • Der Software-Lebenszyklus und Software-Management • Software-Prüfung und Qualitätssicherung • Methoden, Sprachen und Werkzeuge für die einzelnen Phasen: Spezifikation, Grobentwurf, Feinentwurf, Implementierung, Test Viele dieser Aspekte werden speziell mit Bezug auf agile Softwareentwicklung am Beispiel Scrum diskutiert. Dieses Modul kommt, wenn die Voraussetzungen erfüllt sind, auch für andere Fachrichtungen in Frage.		
14. Literatur:	• Ludewig, Lichter: Software Engineering. 2. Aufl. dpunkt-Verlag, 2010 • Pfleeger, Atlee: Software Engineering. Pearson, 2010 • Rubin: Essential Scrum. Addison-Wesley, 2013		
15. Lehrveranstaltungen und -formen:	• 256101 Vorlesung Grundlagen des Software Engineerings • 256102 Übung Grundlagen des Software Engineerings		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiums- /	138 h	
	Nachbearbeitungszeit:		
	Summe:	180 h	
17. Prüfungsnummer/n und -name:	25611 Grundlagen des Software Engineerings (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:	• Folien am Beamer unterstützt durch Tafel und Overhead • Dokumente, Links und Diskussionsforen in ILIAS		

20. Angeboten von: Software-Engineering

Modul: 42410 Grundlagen des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240020	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	• Dirk Pflüger • Stefan Zimmer • Miriam Mehl		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10190 Mathematik für Informatiker und Softwaretechniker und • Modul 10240 Numerische und Stochastische Grundlagen der Informatik bzw. • Modul 41590 Einführung in die Numerik und Stochastik für Softwaretechniker		
12. Lernziele:	Kenntnis grundlegender Konzepte, Algorithmen und Methoden des Wissenschaftlichen Rechnens. Fähigkeit, mit den erlernten Kenntnissen selbständig Methoden zu entwickeln, zu analysieren und umzusetzen, mit denen anwendungsorientierte Probleme effizient und genau gelöst werden können.		
13. Inhalt:	• Überblick über die Simulationspipeline und die Wechselwirkungen zwischen den einzelnen Schritten • Skalenabhängige Modellierung • Diskretisierung (Gitter, Finite Elemente, Zeitschrittverfahren) • Algorithmen (Gittergenerierung, Adaptivität, Lineare Löser, Linked-Cell, Fast Multipole) • Parallelisierung (Gitterpartitionierung, Lastbalancierung) • Kurzer Überblick über die Visualisierung		
14. Literatur:	Martin Hanke-Bourgeois: Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens; Vieweg+Teubner Verlag 2009		
15. Lehrveranstaltungen und -formen:	• 424101 Vorlesung Grundlagen des Wissenschaftlichen Rechnens • 424102 Übung Grundlagen des Wissenschaftlichen Rechnens		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiums- /	138 h	
	Nachbearbeitungszeit:		
	Summe:	180 h	
17. Prüfungsnummer/n und -name:	42411 Grundlagen des Wissenschaftlichen Rechnens (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Simulation großer Systeme		

Modul: 10210 Mensch-Computer-Interaktion

2. Modulkürzel:	051900001	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Albrecht Schmidt		
9. Dozenten:	• Albrecht Schmidt • Thomas Ertl • Daniel Weiskopf • Niels Henze		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10280 Programmierung und Software-Entwicklung		
12. Lernziele:	Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.		
13. Inhalt:	Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet. Die folgenden Themen werden in der Vorlesung behandelt: • Einführung in die Grundlagen der Mensch-Computer Interaktion, historische Entwicklung • Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme • Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers • Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides • Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme • Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen • Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge • Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten • Akzeptanz, Evaluationsmethoden und Qualitätssicherung		
14. Literatur:	• Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin; 2. Auflage. 2010 • Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004 • Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005		
15. Lehrveranstaltungen und -formen:	• 102101 Vorlesung Mensch-Computer-Interaktion		

	• 102102 Übung Mensch-Computer-Interaktion	
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	42 h
	Selbststudiums- /	138 h
	Nachbearbeitungszeit:	
	Summe:	180 h
17. Prüfungsnummer/n und -name:	• 10211 Mensch-Computer-Interaktion (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein	
	• V Vorleistung (USL-V), schriftlich, eventuell mündlich	
18. Grundlage für ... :		
19. Medienform:		
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme	

Modul: 10220 Modellierung

2. Modulkürzel:	052010001	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	• Bernhard Mitschang • Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• 051520005 Programmierung und Software-Entwicklung • 051510005 Datenstrukturen und Algorithmen • 051200005 Systemkonzepte und -programmierung		
12. 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.		
13. Inhalt:	• Entity-Relationship Modell & komplexe Objekte • Relationenmodell & Relationenalgebra , Überblick SQL • Transformationen von ER nach Relationen, Normalisierung • XML, DTD, XML-Schema, Info-Set, Namensräume • Metamodelle & Repository • RDF, RDF-S & Ontologien • UML • Petri Netze, Workflownetze • BPMN		
14. Literatur:	• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002 • R. Eckstein, S. Eckstein, "XML und Datenmodellierung", dpunkt.verlag 2004 • 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 • H.J. Habermann, F. Leymann, "Repository", Oldenbourg 1993 • W. Reisig, "Petri-Netze", Vieweg & Teubner 2010 • B. Silver, "BPMN Method & Style", Cody-Cassidy Press 2009		
15. Lehrveranstaltungen und -formen:	• 102201 Vorlesung Modellierung • 102202 Übung Modellierung		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	• 10221 Modellierung (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein • V Vorleistung (USL-V), schriftlich, eventuell mündlich		
18. Grundlage für ... :	• 10030 Architektur von Anwendungssystemen • 10080 Datenbanken und Informationssysteme		
19. Medienform:			

20. Angeboten von: Institut für Architektur von Anwendungssystemen

400 Fachdidaktikmodule

Zugeordnete Module: 34050 Gestaltung von Lehr- / Lernprozessen im Informatik-Unterricht, Projekt
 34060 Grundlagen der Fachdidaktik Informatik

Modul: 34050 Gestaltung von Lehr- / Lernprozessen im Informatik-Unterricht, Projekt

2. Modulkürzel:	101010072	5. Moduldauer:	1 Semester
3. Leistungspunkte:	4.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	Bernd Zinn		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:			
12. Lernziele:			
13. Inhalt:			
14. Literatur:			
15. Lehrveranstaltungen und -formen:	340501	Seminar Gestaltung von Lehr- / Lernprozessen im Informatik-Unterricht, Projekt	
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	34051	Gestaltung von Lehr- / Lernprozessen im Informatik-Unterricht, Projekt (PL), schriftlich, eventuell mündlich, Gewichtung: 1.0	
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Erziehungswissenschaft		

Modul: 34060 Grundlagen der Fachdidaktik Informatik

2. Modulkürzel:	101010062	5. Moduldauer:	2 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Bernd Zinn		
9. Dozenten:	• Andreas Mußotter • Bernd Zinn		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Es wird empfohlen, die Fachdidaktik möglichst spät zu belegen.		
12. Lernziele:	<i>Die Studierenden sind in der Lage,</i> <i>die Fachdidaktik im Kontext der korrespondierenden Bezugsdisziplinen zu verorten und ihr Bedeutungsspektrum zu überblicken</i> <i>beruflich-technischen Unterricht zielorientiert zu planen und dabei didaktisch-methodische Bezugspunkte kriterienorientiert zu berücksichtigen</i> <i>beruflich-technische Konzepte des Unterrichts so zu gestalten, dass neben fachlich-methodischen auch sozial-kommunikative und personale Kompetenzen unter Berücksichtigung zentraler Aspekte (Umgang mit Inklusion und Heterogenität, Einsatz diagnostischer Verfahren) vermittelt werden können</i> <i>Erkenntnisse aus der (fachdidaktischen) Lehr-Lernforschung im Hinblick auf ihre Bedeutung für das Lehren und Lernen zu interpretieren und diese bei der Konzeptionierung von informationstechnischem Unterricht zu berücksichtigen</i> <i>den komplexen Prozess der Unterrichtsplanung, -durchführung und -evaluation von technischem Unterricht zu erfassen</i> <i>die Durchführung und Evaluation des Unterrichts in ihrer Komplexität als vielfältig interaktiven, inhaltsorientierten und insgesamt kriterienorientierten Prozess zu erfassen und die Ergebnisse kritisch zu reflektieren</i>		
13. Inhalt:	<i>Im Mittelpunkt des Moduls stehen folgende Lerninhalte:</i> • <i>Ausgangslage und Grundkonzeptionen der allgemeinen und beruflichen Technikdidaktik, Stellung der Fachdidaktik im Gefüge der Fachwissenschaft und Erziehungswissenschaft, zentrale Ansätze und Konzepte der beruflichen Bildung</i> • <i>methodisch-didaktische Ansätze im technischen Unterricht, Berufsfeldspezifische Aspekte (z.B. Lernen in technischen Reallernräumen, Experimente)</i> • <i>Umgang mit Inklusion und Heterogenität,</i> • <i>Pädagogische Diagnostik</i> • <i>Analyse berufs- und schulformbezogener Lehrpläne</i> • <i>Planung, Durchführung und Evaluation von technischem Unterricht in der Aus- und Weiterbildung</i>		

- *Wandel beruflicher Anforderungen und Rahmenbedingungen in der Informatik*
- *Aktuelle Inhalte der Lehr-Lernforschung im Bezugsfeld der Technikdidaktik und speziell Fachdidaktik Informatik*

14. Literatur:

- Schubert, S. & Schwill, A. (2011): *Didaktik der Informatik*. 2. Aufl. Spektrum Akademischer Verlag.
- Tenberg, R. (2011): *Vermittlung von fachlichen und überfachlichen Kompetenzen in technischen Berufen. Theorie und Praxis der Technikdidaktik*. Stuttgart: Steiner.
- Nickolaus, R. (2008): *Didaktik - Modelle und Konzepte beruflicher Bildung. Orientierungsleistungen für die Praxis*. Baltmannsweiler: Schneider Hohengehren.
- Nickolaus, R. & Schanz, H. (Hrsg.)(2008): *Didaktik der gewerblich-technischen Berufsbildung*. In: *Diskussion Berufsbildung*, Bd. 9. Baltmannsweiler: Schneider Hohengehren.
- Kuhlmeier, W. (2005): *Berufliche Fachdidaktiken zwischen Anspruch und Realität: Situationsanalyse und Perspektiven einer konzeptionellen Weiterentwicklung am Beispiel der Bereichsdidaktik Bau-, Holz- und Gestaltungstechnik*. Bd. 3. Baltmannsweiler: Schneider Hohengehren.

Aktuelle wissenschaftliche Zeitschriftenbeiträge, insbesondere aus der Lehr-Lernforschung, im Bezugsfeld der beruflichen Technikdidaktik.

15. Lehrveranstaltungen und -formen:

- 340601 Vorlesung Einführung in die Didaktik der Informatik
- 340602 Seminar Vertiefungen zur Einführung in die Didaktik der Informatik

16. Abschätzung Arbeitsaufwand:

- *Präsenzzeit: 2 x 28 h = 56h*
- *Selbststudium: ca. 70 h (Vorlesung)*
- *Selbststudium: ca. 54 h (Seminar)*

Gesamt: ca. 180 h

17. Prüfungsnummer/n und -name:

- 34061 Grundlagen der Fachdidaktik Informatik (PL), schriftlich, eventuell mündlich, Gewichtung: 1.0
- 34062 Grundlagen der Fachdidaktik Informatik, Ausarbeitung incl. Präsentation (USL), schriftlich, eventuell mündlich, Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

3000 Zwischenprüfung

Zugeordnete Module: 10280 Programmierung und Software-Entwicklung
 10940 Theoretische Grundlagen der Informatik
 12060 Datenstrukturen und Algorithmen

Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Daniel Weiskopf		
9. Dozenten:	• Andrés Bruhn • Thomas Ertl • Stefan Funke • Daniel Weiskopf		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	• Modul 10280 Programmierung und Software-Entwicklung		
12. 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. Die Lernziele lassen sich wie folgt zusammenfassen: • 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		
13. Inhalt:	Es werden die folgenden Themen behandelt: • Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen • Komplexität und Effizienz von Algorithmen, O-Notation • Listen (Stack, Queue, doppelt verkettete Listen) • Sortierverfahren (Selection-, Insertion-, Bubble-, Merge-, Quick-Sort) • Bäume (Binär-, AVL-, 2-3-4-, Rot-Schwarz-, B-Bäume, Suchbäume, Traversierung, Heap) • Räumliche Datenstrukturen (uniforme Gitter, Oktal-, BSP-, kD-, CSG-Bäume, Bounding-Volumes) • Graphen (Datenstrukturen, DFS, BFS, topologische Traversierung, Dijkstra-, A*-, Bellman-Ford-Algorithmen, minimale Spannbäume, maximaler Fluss) • Räumliche Graphen (Triangulierung, Voronoi, Delaunay, Graph-Layout) • Textalgorithmen (String-Matching, Knuth-Morris-Pratt, Boyer-Moore, reguläre Ausdrücke, Levenshtein-Distanz) • Hashing (Hashfunktionen, Kollisionen) • Verteilte Algorithmen (Petri-Netze, Programmieren nebenläufiger Abläufe, einige parallele und parallelisierte Algorithmen)		

	• Algorithmenentwurf und -muster (inkrementell, greedy, divide-and-conquer, dynamische Programmierung, Backtracking, randomisierte Algorithmen) • Maschinelles Lernen (überwachtes Lernen, Entscheidungsbäume, SVM, neuronale Netze; unüberwachtes Lernen, k-Means)								
14. Literatur:	• G. Saake, K. Sattler. <i>Algorithmen und Datenstrukturen: Eine Einführung mit Java</i> . 5. Auflage, dpunkt-Verlag, 2013 • T. Ottmann, P. Widmayer. <i>Algorithmen und Datenstrukturen</i> . 5. Auflage, Springer-Verlag, 2012								
15. Lehrveranstaltungen und -formen:	• 120601 Vorlesung Datenstrukturen und Algorithmen • 120602 Übung Datenstrukturen und Algorithmen								
16. Abschätzung Arbeitsaufwand:	<table> <tr> <td>Präsenzzeit:</td><td>63 h</td></tr> <tr> <td>Selbststudiums- /</td><td>207</td></tr> <tr> <td>Nachbearbeitungszeit:</td><td></td></tr> <tr> <td>Summe:</td><td>270 h</td></tr> </table>	Präsenzzeit:	63 h	Selbststudiums- /	207	Nachbearbeitungszeit:		Summe:	270 h
Präsenzzeit:	63 h								
Selbststudiums- /	207								
Nachbearbeitungszeit:									
Summe:	270 h								
17. Prüfungsnummer/n und -name:	• 12061 Datenstrukturen und Algorithmen (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben. • V Vorleistung (USL-V), schriftlich, eventuell mündlich								
18. Grundlage für ... :									
19. Medienform:									
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme								

Modul: 10280 Programmierung und Software-Entwicklung

2. Modulkürzel:	051520005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine. Teilnahme an einem Vorkurs Java ist hilfreich aber nicht notwendig.		
12. Lernziele:	Die Teilnehmer haben einen Überblick über das Gebiet der Informatik. Sie haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	• Die Programmiersprache Java und die virtuelle Maschine • Objekte, Klassen, Schnittstellen, Blöcke, Programmstrukturen, Kontrakte • Klassenmodellierung mit der UML • Objekterzeugung und -ausführung • Boolesche Logik • Verzweigungen, Schleifen, Routinen, Abstraktionen, Modularisierung, Variablen, Zuweisungen • Rechner, Hardware • Syntaxdarstellungen • Übersicht über Programmiersprachen und -werkzeuge • Grundlegende Datenstrukturen und Algorithmen • Vererbung, Polymorphe • Semantik • Programmierung graphischer Oberflächen • Übergang zum Software Engineering		
14. Literatur:	• Appelrath, Hans-Jürgen und Ludewig, Jochen, "Skriptum Informatik - eine konventionelle Einführung", Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999 • Meyer, Bertrand, "Touch of Class", Springer-Verlag, 2009 • Savitch, Walter, "Java. An Introduction to Problem Solving and Programming", Pearson, 6. Auflage, 2012		
15. Lehrveranstaltungen und -formen:	• 102801 Vorlesung Programmierung und Softwareentwicklung • 102802 Übung Programmierung und Softwareentwicklung		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	63 h	
	Selbststudiums- /	187 h	
	Nachbearbeitungszeit:		
	Prüfungsvorbereitung:	20 h	
	Summe:	270 h	

17. Prüfungsnummer/n und -name:	• 10281 Programmierung und Software-Entwicklung (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, • V Vorleistung (USL-V), schriftlich, eventuell mündlich, Vorleistung: Übungsschein. Voraussetzungen werden zu Beginn vom Dozenten festgesetzt. Dazu gehören eine bestimmte Anzahl von Vorträgen in den Übungen und ein bestimmter Teil der Übungspunkte.
18. Grundlage für ... :	12060 Datenstrukturen und Algorithmen
19. Medienform:	• Folien über Beamer • Tafelanschrieb
20. Angeboten von:	Software-Engineering

Modul: 10940 Theoretische Grundlagen der Informatik

2. Modulkürzel:	050420005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	• Volker Diekert • Ulrich Hertrampf		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	Keine		
12. 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.		
13. Inhalt:	Logik und Diskrete Strukturen: • Einführung in die Aussagenlogik: Semantik (Wahrheitswerte), Syntax (Axiome und Schlussregeln), Normalformen; Hornformeln; Endlichkeitssatz; aussagenlogische Resolution; • Einführung in die Prädikatenlogik 1. Stufe: Semantik und Syntax, Normalformen, Unifikatoren, Herbrand-Theorie, prädikatenlogische Resolution; • Elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, Euklidischer Algorithmus, Chinesischer Restsatz, Primzahltests, RSA-Verfahren; Wachstumsabschätzungen; Grundbegriffe der Wahrscheinlichkeitsrechnung; Kombinatorik; Graphen. Automaten und Formale Sprachen: • Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, Iterationslemmata für reguläre und kontextfreie Sprachen, Normalformen, Kellerautomaten, Lösen des Wortproblems kontextfreier Sprachen mit dem CYK-Algorithmus, linear beschränkte Automaten, kontextsensitive Grammatiken, Typ 0-Grammatiken und Turingmaschinen.		
14. Literatur:	• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988 • Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999		
15. 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	
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	84 h
	Selbststudiums- /	276 h
	Nachbearbeitungszeit:	
	Summe:	360 h
17. Prüfungsnummer/n und -name:	• 10941 Theoretische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein • V Vorleistung (USL-V), schriftlich, eventuell mündlich, 30 Min.	
18. Grundlage für ... :		
19. Medienform:		
20. Angeboten von:	Institut für Formale Methoden der Informatik	