



Universität Stuttgart

Modulhandbuch
Studiengang Bachelor of Science Informatik
Prüfungsordnung: 2012

Sommersemester 2015
Stand: 08. April 2015

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:	Apl. Prof. Ulrich Hertrampf Institut für Formale Methoden der Informatik Tel.: 7816-344 E-Mail: ulrich.hertrampf@f05.uni-stuttgart.de
Fachstudienberater/in:	Stefan Zimmer Institut für Parallele und Verteilte Systeme Tel.: E-Mail: stefan.zimmer@ipvs.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	6
Qualifikationsziele	7
100 Basismodule	8
12060 Datenstrukturen und Algorithmen	9
10190 Mathematik für Informatiker und Softwaretechniker	11
10280 Programmierung und Software-Entwicklung	13
10930 Technische Grundlagen der Informatik	15
10940 Theoretische Grundlagen der Informatik	17
200 Kernmodule	19
10020 Algorithmik	20
14910 Berechenbarkeit und Komplexität	22
25610 Grundlagen des Software Engineerings	23
10210 Mensch-Computer-Interaktion	25
10220 Modellierung	27
10240 Numerische und Stochastische Grundlagen	29
36100 Programmierparadigmen	31
41930 Rechnerorganisation	32
40090 Systemkonzepte und -programmierung	34
300 Ergänzungsmodule	36
320 Katalog ISG 1-3	37
10030 Architektur von Anwendungssystemen	38
10060 Computergraphik	40
18560 Grundlagen der Eingebetteten Systeme	42
10100 Grundlagen der Graphischen Ingenieursysteme	43
10110 Grundlagen der Künstlichen Intelligenz	45
56930 Grundlagen der Rechnerarchitektur	47
42410 Grundlagen des Wissenschaftlichen Rechnens	49
10170 Imaging Science	51
330 Katalog ISW 1-3	53
331 Modul anerkannt	54
10030 Architektur von Anwendungssystemen	55
10060 Computergraphik	57
18560 Grundlagen der Eingebetteten Systeme	59
10100 Grundlagen der Graphischen Ingenieursysteme	60
10110 Grundlagen der Künstlichen Intelligenz	62
56930 Grundlagen der Rechnerarchitektur	64
42410 Grundlagen des Wissenschaftlichen Rechnens	66
10170 Imaging Science	68
10180 Information Retrieval und Text Mining	70
39040 Rechnernetze	71
340 Katalog ISW 4	73
31600 Machine learning for NLP	74
350 Wahlmodule aus Master Informatik	75
42910 Advanced Business Process Management	77
55600 Advanced Information Management	79
55740 Advanced Service Computing	81
29550 Algorithmische Geometrie	83

29760	Algorithmische Gruppentheorie	84
45760	Ausgewählte Kapitel der Algorithmentheorie	86
42480	Ausgewählte Kapitel des Wissenschaftlichen Rechnens	87
56680	Automaten über unendlichen Objekten	89
42010	BWL I: Produktion, Organisation, Personalführung, Strategisches Management	91
13200	BWL III: Marketing und Einführung in die Wirtschaftsinformatik	94
10040	Bildsynthese	96
42900	Business Process Management	98
57050	Compilerbau	100
29570	Computer Interface Technologien	102
29430	Computer Vision	104
55640	Correspondence Problems in Computer Vision	106
29580	Data Compression	108
55620	Data Warehousing, Data Mining, and OLAP	109
10080	Datenbanken und Informationssysteme	111
29600	Digital System Design II	113
29590	Digitale Systeme	114
29410	Diskrete Optimierung	116
39250	Distributed Systems I	117
45730	Distributed Systems II	119
57680	Einführung in die Chaostheorie	121
29710	Embedded Systems Engineering	123
58190	Entwurf und Implementierung eines Compilers	125
56980	Fachpraktikum Cloud Architekturen und Management	127
29740	Fachpraktikum Eingebettete Systeme	128
29750	Fachpraktikum Rechnerarchitektur	130
24900	Fachpraktikum Rechnergestützte Methoden der Produktentwicklung	132
45750	Fachpraktikum Verteilte Systeme	134
58440	Fachpraktikum: Algorithmik	135
29440	Geometric Modeling and Computer Animation	136
29450	Graphentheorie	138
46430	Grundlagen der Betriebswirtschaftslehre	140
41980	Grundlagen der VWL	142
29610	Hardware Based Fault Tolerance	144
14380	Hardware Verification and Quality Assessment	146
42920	Hardware-Software-Codesign	148
40620	Hauptseminar INF 1	149
43060	Hauptseminar INF 2	151
42420	High Performance Computing	153
51720	IT-Strategy	155
48500	Image Synthesis	157
51540	Implementierung Finiten Elemente	159
55610	Information Integration	161
55630	Information Visualization and Visual Analytics	163
60120	Interaktive Systeme	165
29420	Konkrete Mathematik	166
29460	Kryptographische Verfahren	168
29480	Loose Coupling and Message Based Applications	170
29470	Machine Learning	172
42220	Marketing I	174
29640	Mikrocontroller	176
29720	Mobile Computing	179
10120	Modellbildung und Simulation	181
29730	Modelling, Simulation, and Specification	183
55650	Multimodal Interaction for Ubiquitous Computers	185
42460	Numerische Simulation	186
40680	Optimization	188
56790	Parallele Numerik	190

29650 Parallele Programmierung	192
10250 Parallele Systeme	193
14740 Partielle Differentialgleichungen (Modellierung, Analysis, Simulation)	195
48570 Practical Course Visual Computing	197
29660 Programmanalysen und Compilerbau	198
51740 Quantencomputing	200
29670 Rapid Prototyping	202
29680 Real-Time Programming	203
29690 Real-Time Video Processing I	205
29700 Real-Time Video Processing II	207
45740 Rechnernetze II	208
48580 Reinforcement Learning	210
36410 Requirements Engineering und Software-Architektur	212
40630 Ringvorlesung Informatik	213
48600 Robotics I	215
48610 Robotics II	217
48620 Scientific Visualization	219
29510 Service Computing	221
31080 Service Engineering	223
46660 Service Management and Cloud Computing, and Evaluation	225
42520 Services and Service Composition	227
14480 Sichere und zuverlässige Softwaresysteme	229
16500 Software Engineering	230
56550 Software Verification	231
42810 Software-Qualitätssicherung und -Wartung	233
60140 Sprachbau mit Language Workbenches	234
48640 Theoretical and Methodological Foundations of Autonomous Systems	235
46450 Vertiefungslinie Architektur von Anwendungssystemen	237
29330 Vertiefungslinie Datenbanken und Informationssysteme	238
29340 Vertiefungslinie Intelligent Systems	240
46470 Vertiefungslinie Parallele Systeme	242
46440 Vertiefungslinie Programmiersprachen, Compilerbau, Programmanalysen	243
29370 Vertiefungslinie Rechnerarchitekturen und eingebettete Systeme	245
46460 Vertiefungslinie Sprachverarbeitung	247
29380 Vertiefungslinie Theoretische Informatik und Wissenschaftliches Rechnen	249
29390 Vertiefungslinie Verteilte Systeme	251
29400 Vertiefungslinie Visualisierung und Interaktive Systeme	253
29500 Visual Computing	255
34940 Weiterführende Numerik partieller Differentialgleichungen	257
400 Schlüsselqualifikationen fachaffin	258
10290 Projekt-INF	259
38610 Seminar-INF 1	261
42400 Seminar-INF 2	262
81100 Bachelorarbeit Informatik	263

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.

Auf den Bachelor-Studiengang Informatik bauen die Masterstudiengänge auf, die vom Fachbereich Informatik im Anschluss an das Bachelor-Studium angeboten werden. Dazu gehören neben den deutschsprachigen Masterstudiengängen *Informatik* und *Softwaretechnik* auch die englischsprachigen Masterstudiengänge *Computer Science* und *Computational Linguistics*. Es wird empfohlen, den Master als Abschluss eines universitären Studiums anzustreben.

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

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

Qualifikationsziele

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

Die Absolventinnen und Absolventen des Bachelorstudienganges Informatik

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

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

100 Basismodule

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

Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Lars Grunske		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Lars Grunske • Stefan Funke 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 2. Semester → Pflichtmodule</p> <p>KLAGymPO Informatik, PO 2010, 2. Semester → Pflichtmodule</p> <p>B.Sc. Informatik, PO 2009, 2. Semester → Basismodule</p> <p>B.Sc. Informatik, PO 2012, 2. Semester → Basismodule</p> <p>BA (Komb) Informatik, PO 2009, 2. Semester → Module im Nebenfach</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051520005 Programmierung und Software-Entwicklung 		
12. Lernziele:	<p>Die Studierenden kennen nach engagierter Mitarbeit in dieser Veranstaltung diverse zentrale Algorithmen auf geeigneten Datenstrukturen, die für eine effiziente Nutzung von Computern unverzichtbar sind. Sie können am Ende zu gängigen Problemen geeignete programmiersprachliche Lösungen angeben und diese in einer konkreten Programmiersprache formulieren.</p> <p>Konkret:</p> <ul style="list-style-type: none"> • 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 sequentieller Algorithmen 		
13. Inhalt:	<ul style="list-style-type: none"> • Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen • Komplexität und Effizienz von Algorithmen, O-Notation • Wahl der Datenstrukturen; Listen, Bäume, Graphen; deren Definitionen, deren Datenstrukturen • 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) • diverse Graphenalgorithmen (DFS, BFS, Besuchssequenzen, topol. Traversierung, Zusammenhangskomponenten, minimale Spannbäume, Dijkstra-, Floyd- kürzeste Wege) • Algorithmen auf Mengen und Relationen (transitive Hüllen, Warshall) 		

	<ul style="list-style-type: none">• Korrektheitsbegriff und -formalismen; Spezifikation und Implementierung• Einige parallele und parallelisierte Algorithmen• einfache Elemente paralleler Programmierung, soweit für obiges notwendig
14. Literatur:	<ul style="list-style-type: none">• Appelrath H.J., Ludewig. J., Skriptum Informatik, 1999• Sedgewick, R., Algorithms in C, 1998
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 120601 Vorlesung Datenstrukturen und Algorithmen• 120602 Übung Datenstrukturen und Algorithmen
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 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:	

Modul: 10190 Mathematik für Informatiker und Softwaretechniker

2. Modulkürzel:	080300100	5. Moduldauer:	2 Semester
3. Leistungspunkte:	18.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	12.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Wolfgang Rump		
9. Dozenten:	<ul style="list-style-type: none"> • Peter Lesky • Wolfgang Rump • Wolf-Patrick Düll • Andreas Markus Kollross 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 3. Semester → Pflichtmodule</p> <p>KLAgymPO Informatik, PO 2010, 3. Semester → Pflichtmodule</p> <p>B.Sc. Informatik, PO 2009, 1. Semester → Basismodule</p> <p>B.Sc. Informatik, PO 2012, 1. Semester → Basismodule</p>		
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:	<p>1. Semester:</p> <ul style="list-style-type: none"> • 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) <p>2. Semester:</p> <ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • Anna Sändig, Mathematik, Vorlesungskripte , SS 2007 • D. Hachenberger, Mathematik für Informatiker, 2005 • M. Brill, Mathematik für Informatiker, 2001 • P.Hartmann, Mathematik für Informatiker, 2002 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 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: 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. Stefan Wagner		
9. Dozenten:	Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:	LAGymPO Informatik, PO 2010, 1. Semester → Pflichtmodule KLAGymPO Informatik, PO 2010, 1. Semester → Pflichtmodule B.Sc. Informatik, PO 2009, 1. Semester → Basismodule B.Sc. Informatik, PO 2012, 1. Semester → Basismodule BA (Komb) Informatik, PO 2009, 3. Semester → Module im Nebenfach		
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:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 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 		

	<ul style="list-style-type: none"> • Savitch, Walter, "Java. An Introduction to Problem Solving and Programming", Pearson, 6. Auflage, 2012
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102801 Vorlesung Programmierung und Softwareentwicklung • 102802 Übung Programmierung und Softwareentwicklung
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Vor-/Nachbearbeitungszeit: 187 Stunden Prüfungsvorbereitung: 20 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • Folien über Beamer • Tafelanschrieb
20. Angeboten von:	Software-Engineering

Modul: 10930 Technische Grundlagen der Informatik

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

Nachbearbeitungszeit: 117 Stunden

-
17. Prüfungsnummer/n und -name:
- 10931 Technische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: In dem Fach Elektrotechnische Grundlagen werden ein oder zwei Klausuren Semester-begleitend durchgeführt. Es ist eine Mindestzahl von Punkten aus dieser Prüfung bzw. diesen Prüfungen erforderlich, um zur Prüfung Technische Grundlagen der Informatik zugelassen zu werden. Bezüglich der Elektrotechnische Grundlagen und den Digitaltechnische Komponenten ist eine Teilnahme an einer Mindestzahl der Übungen, die zu Beginn der Lehrveranstaltung festgelegt wird, erforderlich.
 - V Vorleistung (USL-V), schriftlich, eventuell mündlich
-
18. Grundlage für ... :
-
19. Medienform:
-
20. Angeboten von: Eingebettete Systeme (Embedded Systems 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:	<ul style="list-style-type: none"> • Volker Diekert • Ulrich Hertrampf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 1. Semester → Pflichtmodule</p> <p>KLAgymPO Informatik, PO 2010, 1. Semester → Pflichtmodule</p> <p>B.Sc. Informatik, PO 2009, 1. Semester → Basismodule</p> <p>B.Sc. Informatik, PO 2012, 1. Semester → Basismodule</p>		
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	<p>Logik und Diskrete Strukturen:</p> <ul style="list-style-type: none"> • 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. <p>Automaten und Formale Sprachen:</p> <ul style="list-style-type: none"> • 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:	<p>Logik und Diskrete Strukturen:</p> <ul style="list-style-type: none"> • 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. <p>Automaten und Formale Sprachen:</p> <ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none">• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988• Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 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 Stunden Nachbearbeitungszeit: 276 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 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

200 Kernmodule

Zugeordnete Module:

- 10020 Algorithmmik
- 10210 Mensch-Computer-Interaktion
- 10220 Modellierung
- 10240 Numerische und Stochastische Grundlagen
- 14910 Berechenbarkeit und Komplexität
- 25610 Grundlagen des Software Engineerings
- 36100 Programmierparadigmen
- 40090 Systemkonzepte und -programmierung
- 41930 Rechnerorganisation

Modul: 10020 Algorithmik

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

20. Angeboten von:

Institut für Formale Methoden der Informatik

Modul: 14910 Berechenbarkeit und Komplexität

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

Modul: 25610 Grundlagen des Software Engineerings

2. Modulkürzel:	51520170	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Stefan Wagner		
9. Dozenten:	Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:	LAGymPO Informatik, PO 2010, 5. Semester → Wahlmodule B.Sc. Informatik, PO 2009, 5. Semester → Kernmodule B.Sc. Informatik, PO 2012, 5. Semester → Kernmodule M.Sc. Informatik, PO 2012, 1. Semester → Auflagenmodule des Masters		
11. Empfohlene Voraussetzungen:	051520005 Programmierung und Software-Entwicklung 051510005 Datenstrukturen und Algorithmen sowie entsprechende Programmiererfahrung		
12. Lernziele:	Die Teilnehmer kennen die Grundbegriffe des Software Engineerings und haben einen Überblick über die Methoden und Techniken, die dort angewandt werden. Einige ausgewählte Methoden und Techniken können angewandt werden.		
13. Inhalt:	<p>Software Engineering kann in einer Vorlesung nicht erschöpfend behandelt werden. GSE gibt einen Überblick über das Gebiet und vertieft einzelne Themen, damit diese in der Praxis verwendet werden können. Es bildet damit auch die Basis für weitere Vertiefungen in diesem Gebiet. Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung. Die einzelnen Themen sind:</p> <ul style="list-style-type: none"> • 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 <p>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.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Ludwig, Licher: Software Engineering. 2. Aufl. dpunkt-Verlag, 2010 • Pfleeger, Atlee: Software Engineering. Pearson, 2010 • Rubin: Essential Scrum. Addison-Wesley, 2013 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 256101 Vorlesung Grundlagen des Software Engineerings • 256102 Übung Grundlagen des Software Engineerings 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138		
17. Prüfungsnummer/n und -name:	25611 Grundlagen des Software Engineerings (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		

18. Grundlage für ... :

19. Medienform:

- Folien am Beamer unterstützt durch Tafel und Overhead
- Dokumente, Links und Diskussionsforen in ILIAS

20. Angeboten von:

Software-Engineering

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:	<ul style="list-style-type: none"> • Albrecht Schmidt • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 6. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Kernmodule</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051200005 Systemkonzepte und -programmierung 		
12. Lernziele:	<p>Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.</p>		
13. Inhalt:	<p>Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet.</p> <p>Die folgenden Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> • 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 		

	<ul style="list-style-type: none">• Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten• Akzeptanz, Evaluationsmethoden und Qualitätssicherung
14. Literatur:	<ul style="list-style-type: none">• 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:	<ul style="list-style-type: none">• 102101 Vorlesung Mensch-Computer-Interaktion• 102102 Übung Mensch-Computer-Interaktion
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 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:	<ul style="list-style-type: none"> • Bernhard Mitschang • Frank Leymann 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Kernmodule</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Kernmodule</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 102201 Vorlesung Modellierung • 102202 Übung Modellierung 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden		

Nachbearbeitungszeit: 138 Stunden

17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 10221 Modellierung (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	<ul style="list-style-type: none">• 10030 Architektur von Anwendungssystemen• 10080 Datenbanken und Informationssysteme
19. Medienform:	
20. Angeboten von:	Institut für Architektur von Anwendungssystemen

Modul: 10240 Numerische und Stochastische Grundlagen

2. Modulkürzel:	051240005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 3. Semester → Kernmodule B.Sc. Informatik, PO 2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10190 Mathematik für Informatiker und Softwaretechniker 		
12. Lernziele:	Beherrschung grundlegender Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der erlernten Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.		
13. Inhalt:	Methoden der angewandten Mathematik, insbesondere der Numerik, Stochastik und Statistik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse: <ul style="list-style-type: none"> • numerische Algorithmik • Gleitpunktzahlen und Gleitpunktarithmetik • Interpolation & Approximation • Integration • lineare Gleichungssysteme • Iterative Lösung linearer und nichtlinearer Gleichungen • gewöhnliche Differentialgleichungen • Stochastik • Zufall und Unsicherheit • diskrete und kontinuierliche Wahrscheinlichkeitsräume • Asymptotik • Elementare induktive Statistik Dabei wird ein konstruktiv-algorithmischer Zugang gewählt, der sich an konkreten Aufgabenstellungen aus der Informatik orientiert.		
14. Literatur:	<ul style="list-style-type: none"> • Huckle, Schneider; Numerik für Informatiker • Schickinger T., Steger A.; Diskrete Strukturen, Band 2, 2002 • Dahmen, Reusken; Numerik für Ingenieure 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102401 Vorlesung Numerische und Stochastische Grundlagen der Informatik • 102402 Übung Numerische und Stochastische Grundlagen der Informatik 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden		

17. Prüfungsnummer/n und -name:

- 10241 Numerische und Stochastische Grundlagen (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: Simulation großer Systeme

Modul: 36100 Programmierparadigmen

2. Modulkürzel:	051510010	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. Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 4. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	Programmiererfahrung in mindestens einer Programmiersprache, vornehmlich Java, so wie z. B. im Modul "Programmierung und Software Entwicklung" (10280) erworben.		
12. Lernziele:	Die Studierenden haben grundlegende Konzepte von Programmiersprachen verstanden, die dem Erlernen weiterer Sprachen und dem vertieften Verständnis ihnen bekannter Sprachen dienlich sind. Sie haben deren Anwendung in mindestens einer weiteren Programmiersprache ihrer Ausbildung verstanden. Sie können ihre Kenntnisse in einfachen Programmen anwenden. Sie können weitere Programmiersprachen in ihrer akademischen und beruflichen Karriere schneller und präziser erlernen.		
13. Inhalt:	Überblick typischer Konzepte in Programmiersprachen und ihrer Auswirkungen auf die Sprache und deren Anwendung: Grundsätzliche Ausführungsmodelle, Speichermodelle und deren Konsequenzen, Datentypen und Typsysteme, unterschiedliche Bindungskonzepte und ihre Auswirkungen, objekt-orientierte Sprachkonzepte, Abstraktion und Kompositionsmechanismen, funktionale Sprachen. Eventuell werden auch Elemente der parallelen Programmierung und der Logik-Programmierung mit einbezogen. Die Vorlesung ist kein Streifzug durch diverse Programmiersprachen, sondern die Vorstellung zugrundeliegender Prinzipien, und ihrer Begründung aus der Sicht des Software Engineering, insbesondere der Zuverlässigkeit der Anwendung, und, wo nötig, der Implementierungsmodelle.		
14. Literatur:	<ul style="list-style-type: none"> • Sebesta, Robert, Concepts of Programming Languages, Pearson Verlag, 2010 (Hörerschein verfügbar) • weitere Literatur wird zu Beginn der Lehrveranstaltung und auf den Webseiten bekanntgegeben. 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 361001 Vorlesung Programmierparadigmen • 361002 Übung Programmierparadigmen 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 36101 Programmierparadigmen (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Softwaretechnologie		

Modul: 41930 Rechnerorganisation

2. Modulkürzel:	051700005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	9.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Martin Radetzki • Sven Simon 		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Für Rechnerorganisation 1: Einführung in die Technische Informatik (14360) • Für Rechnerorganisation 2: Erfolgreiche Teilnahme an den Übungen von Rechnerorganisation 1 oder eine bestandene Eingangsklausur 		
12. Lernziele:	<p>Rechnerorganisation 1:</p> <ul style="list-style-type: none"> • Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung • Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen <p>Rechnerorganisation 2:</p> <ul style="list-style-type: none"> • Kenntnisse in den Grundlagen des Entwurfs digitaler Schaltungen und Systeme, • Fertigkeiten in der Verwendung von Hardware-Beschreibungssprachen • Fertigkeiten im Umgang mit programmierbarer Logik (FPGA) und Prototypenboards, • Fertigkeiten im Umgang mit Werkzeugen zur Entwurfsautomatisierung, • Kenntnisse des Zusammenhangs von Hard- und Software • Erfahrung in Projektarbeit im Team 		
13. Inhalt:	<p>Rechnerorganisation 1:</p> <ul style="list-style-type: none"> • Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Experimente mit Prozessorsimulatoren vertieft. <p>Im einzelnen werden behandelt:</p> <ul style="list-style-type: none"> • 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

Rechnerorganisation 2:

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

14. Literatur:	Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 419301 Vorlesung Rechnerorganisation 1 • 419302 Übung Rechnerorganisation 1 • 419303 Vorlesung Rechnerorganisation 2 • 419304 Hardwarepraktikum Rechnerorganisation 2
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 95 Stunden Nachbearbeitungszeit: 265 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 41931 Rechnerorganisation (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Rechnerarchitektur

Modul: 40090 Systemkonzepte und -programmierung

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

- Höhere Kommunikationskonzepte

Basialgorithmen für Verteilte Systeme

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

Praktische nebenläufige Programmierung in Java

- Threads und Synchronisation
- Socketschnittstelle
- RMI Programmierung

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

300 Ergänzungsmodule

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

320 Katalog ISG 1-3

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10060	Computergraphik
	10100	Grundlagen der Graphischen Ingenieursysteme
	10110	Grundlagen der Künstlichen Intelligenz
	10170	Imaging Science
	18560	Grundlagen der Eingebetteten Systeme
	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:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	<p>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.</p>		
13. Inhalt:	<p>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.</p>		
14. Literatur:	<ul style="list-style-type: none"> • 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 		

	<ul style="list-style-type: none"> • M. Fowler, Patters 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:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 10031 Architektur von Anwendungssystemen (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	<ul style="list-style-type: none"> • 29480 Loose Coupling and Message Based Applications • 29490 Services und Service Komposition • 29510 Service Computing • 29530 Business Process Management
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:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Martin Fuchs • Guido Reina 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051900001 Mensch-Computer-Interaktion • Modul 051240005 Numerik und Stochastik. 		
12. Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.		
13. Inhalt:	<p>Folgende Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> • 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 Verdeckungsrechnung • Grundlagen der geometrischen Modellierung (Kurven, Flächen) • Räumliche Datenstrukturen <p>Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.</p>		
14. Literatur:	<ul style="list-style-type: none"> • J. Encarnacao, W. Strasser, R. Klein, Graphische Datenverarbeitung (Band1 und 2), 1997 		

	<ul style="list-style-type: none">• J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 100601 Vorlesung Computergraphik• 100602 Übung Computergraphik
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 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: 18560 Grundlagen der Eingebetteten Systeme

2. Modulkürzel:	051711010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 → B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →		
11. Empfohlene Voraussetzungen:	Modul 051700005 Rechnerorganisation		
12. Lernziele:	Methodische und werkzeugunterstützte Konstruktion von Hardware und Software für auf spezifische Anwendungen hin optimierte eingebettete Systeme.		
13. Inhalt:	Einführung in Charakteristik und Anwendungsgebiete eingebetteter Systeme, hardwarenahe Softwareentwicklung, Software-Scheduling, eingebettete Algorithmen (digitale Signalverarbeitung, Kanal- und Quellencodierung am Beispiel Viterbi-Algorithmus und MPEG-Video-Codec), zustandsbasierte Modellierung (Statecharts), eingebettete Prozessoren (Microcontroller, digitale Signalprozessoren, ARM), Bussysteme und Speicher, Systemsynthese (Taskgraphen, Allokation, Bindung, Ablaufplanung/Scheduling)		
14. Literatur:	J. Teich, Digitale Hardware/Software-Systeme, 2. Auflage, 2007 P. Marwedel, Embedded System Design, 2006		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 185601 Vorlesung Grundlagen der Eingebetteten Systeme • 185602 Übung Grundlagen der Eingebetteten Systeme 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 18561 Grundlagen der Eingebetteten Systeme (PL), schriftliche Prüfung, 90 Min., Gewichtung: 75.0 • V Vorleistung (USL-V), schriftliche Prüfung 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Eingebettete Systeme (Embedded Systems Engineering)		

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:	<p>LAGymPO Informatik, PO 2010, 4. Semester → Ergänzendes Modul</p> <p>LAGymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>KLAgymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	Grundstudium		
12. Lernziele:	<ul style="list-style-type: none"> • Kenntnis und Verständnis von Modellen bei der Produktentwicklung • Grundkenntnisse über die wichtigsten Modellarten, Algorithmen und Datenstrukturen und Techniken für den Datenaustausch 		
13. Inhalt:	<p>Inhalte:</p> <ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 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 Stunden
 Nachbearbeitungszeit: 138 Stunden

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:

Modul: 10110 Grundlagen der Künstlichen Intelligenz

2. Modulkürzel:	051900205	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. Marc Toussaint		
9. Dozenten:	<ul style="list-style-type: none"> • Andrés Bruhn • Marc Toussaint 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 080300100 Mathematik für Informatiker und Softwaretechniker 		
12. 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.		
13. Inhalt:	<ul style="list-style-type: none"> • 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 		
14. Literatur:	<ul style="list-style-type: none"> • S. Russell, P. Norvig, Künstliche Intelligenz, 2004 • G. F. Luger, Künstliche Intelligenz, 2001 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101101 Vorlesung Grundlagen der Künstlichen Intelligenz • 101102 Übung Grundlagen der Künstlichen Intelligenz 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

Gesamt: 180 Stunden

17. Prüfungsnummer/n und -name:

- 10111 Grundlagen der Künstlichen Intelligenz (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben
- V Vorleistung (USL-V), schriftlich, eventuell mündlich

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Visualisierung und Interaktive Systeme

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:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Rafal Baranowski 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 4. Semester → Ergänzendes Modul</p> <p>LAGymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>KLAgymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 41930 Rechnerorganisation 		
12. Lernziele:	<ul style="list-style-type: none"> • Tiefes Verständnis von Entwurfskonzepten, die in modernen Prozessoren und Rechensystemen Verwendung finden • Kenntnis von Entwurfsherausforderungen. • Verständnis von aktuellen und zukünftigen Entwicklungstrends 		
13. Inhalt:	<p>Grundlegende und fortgeschrittene Themen der Rechnerarchitektur, inklusive:</p> <ul style="list-style-type: none"> • 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. 		

	<ul style="list-style-type: none">• Datenparallelismus (DLP): Vektorprozessoren, SIMD, Grafikprozessoreinheiten (GPGPU)• Threadparallelismus (TLP): Mehrprozessorsysteme, Speicherkohärenz und Synchronisierung.• Speicher- und Cache-Architekturen: Entwurf und Optimierung
14. Literatur:	<ul style="list-style-type: none">• 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:	<ul style="list-style-type: none">• 569301 Vorlesung Grundlagen der Rechnerarchitektur• 569302 Übung Grundlagen der Rechnerarchitektur
16. Abschätzung Arbeitsaufwand:	<ul style="list-style-type: none">• Präsenzzeit: 42 Stunden• Selbststudium: 138 Stunden• Summe: 180 Stunden
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: 42410 Grundlagen des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240020	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 6. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<p>080300100 Mathematik für Informatiker und Softwaretechniker und 051240005 Numerische und Stochastische Grundlagen der Informatik bzw. 051240006 Einführung in die Numerik und Stochastik für Softwaretechniker</p>		
12. Lernziele:	<p>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.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • 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:	<p>Martin Hanke-Bourgeois: Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens; Vieweg+Teubner Verlag 2009</p>		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 424101 Vorlesung Grundlagen des Wissenschaftlichen Rechnens• 424102 Übung Grundlagen des Wissenschaftlichen Rechnens
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudiumszeit: 138 Stunden
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: 10170 Imaging Science

2. Modulkürzel:	051900210	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10190 Mathematik für Informatiker und Softwaretechniker 		
12. Lernziele:	<p>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.</p> <p>The student knows the basics of digital image representation and processing and is able to solve problems of the field using the methods presented in the course.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • Grundlagen aus der Optik:Lochkamera, Linsengleichung • Bildaufnahme:Kameras, Objektive, Beleuchtung, Aufnahmeprozess • Bildrepräsentation:Diskretisierung, Farbräume • Elementare Bildbearbeitung:Punktoperationen (z.B. Kontrastverstärkung, Binarisierung) • Lineare und nichtlineare Filter:Faltung, morphologische Operatoren • Fouriertransformation, Bilddarstellung und -bearbeitung im Fourierraum, Abtasttheorem • Orthogonale Transformationen:Cosinus, Wavelets • Kompression:Generische Verfahren (RLE, Entropie), spezielle Bildverfahren (z.B. jpeg) • Video:Formate, Kompression (z.B. MPEG) • Bildverbesserung und Restauration • Elementare Segmentierungsverfahren <ul style="list-style-type: none"> • Fundamentals of optics such as pinhole camera and lens equation • Image acquisition: Cameras, lenses, illumination, acquisition process 		

	<ul style="list-style-type: none"> • Image representation: Discretization, color spaces • Basics of image processing, e.g. point operations such as contrast enhancement or binarization • Linear and nonlinear filtering such as convolution and morphological operations. • Fourier transform, image representation and processing in Fourier space, sampling theorem • Orthogonal transforms such as cosine transform and wavelets • Compression: Generic compression (RLE, entropy coding), methods specialized to domain of images (e.g. jpeg) • Video: file formats, compression (e.g. mpeg) • Image enhancement and restauration • Basics of segmentation
14. Literatur:	<ul style="list-style-type: none"> • Bässmann, Henning; Kreys, Jutta, Bildverarbeitung Ad Oculos, 2004 • 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 • Bigun, J.: Vision with Direction, 2006 • Klaus D. Tönnies, Grundlagen der Bildverarbeitung, 2005 • L. G. Shapiro, G. C. Stockman, Computer Vision, 2001
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101701 Vorlesung Imaging Science • 101702 Übung Imaging Science
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden</p> <p>Gesamt: 180 Stunden</p>
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10171 Imaging Science (PL), schriftlich, eventuell mündlich, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben. • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	<ul style="list-style-type: none"> • 29430 Computer Vision • 55640 Correspondence Problems in Computer Vision
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

330 Katalog ISW 1-3

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10060	Computergraphik
	10100	Grundlagen der Graphischen Ingenieursysteme
	10110	Grundlagen der Künstlichen Intelligenz
	10170	Imaging Science
	10180	Information Retrieval und Text Mining
	18560	Grundlagen der Eingebetteten Systeme
	331	Modul anerkannt
	39040	Rechnernetze
	42410	Grundlagen des Wissenschaftlichen Rechnens
	56930	Grundlagen der Rechnerarchitektur

331 Modul anerkannt

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:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	<p>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.</p>		
13. Inhalt:	<p>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.</p>		
14. Literatur:	<ul style="list-style-type: none"> • 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 		

	<ul style="list-style-type: none"> • M. Fowler, Patters 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:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 10031 Architektur von Anwendungssystemen (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	<ul style="list-style-type: none"> • 29480 Loose Coupling and Message Based Applications • 29490 Services und Service Komposition • 29510 Service Computing • 29530 Business Process Management
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:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Martin Fuchs • Guido Reina 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051900001 Mensch-Computer-Interaktion • Modul 051240005 Numerik und Stochastik. 		
12. Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.		
13. Inhalt:	<p>Folgende Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> • 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 Verdeckungsrechnung • Grundlagen der geometrischen Modellierung (Kurven, Flächen) • Räumliche Datenstrukturen <p>Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.</p>		
14. Literatur:	<ul style="list-style-type: none"> • J. Encarnacao, W. Strasser, R. Klein, Graphische Datenverarbeitung (Band1 und 2), 1997 		

	<ul style="list-style-type: none">• J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 100601 Vorlesung Computergraphik• 100602 Übung Computergraphik
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 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: 18560 Grundlagen der Eingebetteten Systeme

2. Modulkürzel:	051711010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 → B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →		
11. Empfohlene Voraussetzungen:	Modul 051700005 Rechnerorganisation		
12. Lernziele:	Methodische und werkzeugunterstützte Konstruktion von Hardware und Software für auf spezifische Anwendungen hin optimierte eingebettete Systeme.		
13. Inhalt:	Einführung in Charakteristik und Anwendungsgebiete eingebetteter Systeme, hardwarenahe Softwareentwicklung, Software-Scheduling, eingebettete Algorithmen (digitale Signalverarbeitung, Kanal- und Quellencodierung am Beispiel Viterbi-Algorithmus und MPEG-Video-Codec), zustandsbasierte Modellierung (Statecharts), eingebettete Prozessoren (Microcontroller, digitale Signalprozessoren, ARM), Bussysteme und Speicher, Systemsynthese (Taskgraphen, Allokation, Bindung, Ablaufplanung/Scheduling)		
14. Literatur:	J. Teich, Digitale Hardware/Software-Systeme, 2. Auflage, 2007 P. Marwedel, Embedded System Design, 2006		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 185601 Vorlesung Grundlagen der Eingebetteten Systeme • 185602 Übung Grundlagen der Eingebetteten Systeme 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 18561 Grundlagen der Eingebetteten Systeme (PL), schriftliche Prüfung, 90 Min., Gewichtung: 75.0 • V Vorleistung (USL-V), schriftliche Prüfung 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Eingebettete Systeme (Embedded Systems Engineering)		

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:	LAGymPO Informatik, PO 2010, 4. Semester → Ergänzendes Modul LAGymPO Informatik, PO 2010, 4. Semester → Wahlmodule KLAGymPO Informatik, PO 2010, 4. Semester → Wahlmodule B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 → B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 → B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 → B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 → BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG → BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →		
11. Empfohlene Voraussetzungen:	Grundstudium		
12. Lernziele:	<ul style="list-style-type: none"> • Kenntnis und Verständnis von Modellen bei der Produktentwicklung • Grundkenntnisse über die wichtigsten Modellarten, Algorithmen und Datenstrukturen und Techniken für den Datenaustausch 		
13. Inhalt:	Inhalte: <ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • D. Roller, CAD - Effiziente Anpassungs- und Variantenkonstruktion, Springer-Verlag 		

	<ul style="list-style-type: none">• Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 101001 Vorlesung Grundlagen der Graphischen Ingenieursysteme• 101002 Übung Grundlagen der Graphischen Ingenieursysteme
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
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:	

Modul: 10110 Grundlagen der Künstlichen Intelligenz

2. Modulkürzel:	051900205	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. Marc Toussaint		
9. Dozenten:	<ul style="list-style-type: none"> • Andrés Bruhn • Marc Toussaint 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 080300100 Mathematik für Informatiker und Softwaretechniker 		
12. 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.		
13. Inhalt:	<ul style="list-style-type: none"> • 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 		
14. Literatur:	<ul style="list-style-type: none"> • S. Russell, P. Norvig, Künstliche Intelligenz, 2004 • G. F. Luger, Künstliche Intelligenz, 2001 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101101 Vorlesung Grundlagen der Künstlichen Intelligenz • 101102 Übung Grundlagen der Künstlichen Intelligenz 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

Gesamt: 180 Stunden

17. Prüfungsnummer/n und -name:

- 10111 Grundlagen der Künstlichen Intelligenz (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben
- V Vorleistung (USL-V), schriftlich, eventuell mündlich

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Visualisierung und Interaktive Systeme

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:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Rafal Baranowski 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 4. Semester → Ergänzendes Modul</p> <p>LAGymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>KLAgymPO Informatik, PO 2010, 4. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 41930 Rechnerorganisation 		
12. Lernziele:	<ul style="list-style-type: none"> • Tiefes Verständnis von Entwurfskonzepten, die in modernen Prozessoren und Rechensystemen Verwendung finden • Kenntnis von Entwurfsherausforderungen. • Verständnis von aktuellen und zukünftigen Entwicklungstrends 		
13. Inhalt:	<p>Grundlegende und fortgeschrittene Themen der Rechnerarchitektur, inklusive:</p> <ul style="list-style-type: none"> • 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. 		

	<ul style="list-style-type: none"> • Datenparallelismus (DLP): Vektorprozessoren, SIMD, Grafikprozessoreinheiten (GPGPU) • Threadparallelismus (TLP): Mehrprozessorsysteme, Speicherkohärenz und Synchronisierung. • Speicher- und Cache-Architekturen: Entwurf und Optimierung
14. Literatur:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • 569301 Vorlesung Grundlagen der Rechnerarchitektur • 569302 Übung Grundlagen der Rechnerarchitektur
16. Abschätzung Arbeitsaufwand:	<ul style="list-style-type: none"> • Präsenzzeit: 42 Stunden • Selbststudium: 138 Stunden • Summe: 180 Stunden
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: 42410 Grundlagen des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240020	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 6. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<p>080300100 Mathematik für Informatiker und Softwaretechniker und 051240005 Numerische und Stochastische Grundlagen der Informatik bzw. 051240006 Einführung in die Numerik und Stochastik für Softwaretechniker</p>		
12. Lernziele:	<p>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.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • 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:	<p>Martin Hanke-Bourgeois: Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens; Vieweg+Teubner Verlag 2009</p>		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 424101 Vorlesung Grundlagen des Wissenschaftlichen Rechnens• 424102 Übung Grundlagen des Wissenschaftlichen Rechnens
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudiumszeit: 138 Stunden
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: 10170 Imaging Science

2. Modulkürzel:	051900210	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISG 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10190 Mathematik für Informatiker und Softwaretechniker 		
12. Lernziele:	<p>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.</p> <p>The student knows the basics of digital image representation and processing and is able to solve problems of the field using the methods presented in the course.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • Grundlagen aus der Optik:Lochkamera, Linsengleichung • Bildaufnahme:Kameras, Objektive, Beleuchtung, Aufnahmeprozess • Bildrepräsentation:Diskretisierung, Farbräume • Elementare Bildbearbeitung:Punktoperationen (z.B. Kontrastverstärkung, Binarisierung) • Lineare und nichtlineare Filter:Faltung, morphologische Operatoren • Fouriertransformation, Bilddarstellung und -bearbeitung im Fourierraum, Abtasttheorem • Orthogonale Transformationen:Cosinus, Wavelets • Kompression:Generische Verfahren (RLE, Entropie), spezielle Bildverfahren (z.B. jpeg) • Video:Formate, Kompression (z.B. MPEG) • Bildverbesserung und Restauration • Elementare Segmentierungsverfahren <ul style="list-style-type: none"> • Fundamentals of optics such as pinhole camera and lens equation • Image acquisition: Cameras, lenses, illumination, acquisition process 		

	<ul style="list-style-type: none"> • Image representation: Discretization, color spaces • Basics of image processing, e.g. point operations such as contrast enhancement or binarization • Linear and nonlinear filtering such as convolution and morphological operations. • Fourier transform, image representation and processing in Fourier space, sampling theorem • Orthogonal transforms such as cosine transform and wavelets • Compression: Generic compression (RLE, entropy coding), methods specialized to domain of images (e.g. jpeg) • Video: file formats, compression (e.g. mpeg) • Image enhancement and restauration • Basics of segmentation
14. Literatur:	<ul style="list-style-type: none"> • Bässmann, Henning; Kreys, Jutta, Bildverarbeitung Ad Oculos, 2004 • 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 • Bigun, J.: Vision with Direction, 2006 • Klaus D. Tönnies, Grundlagen der Bildverarbeitung, 2005 • L. G. Shapiro, G. C. Stockman, Computer Vision, 2001
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101701 Vorlesung Imaging Science • 101702 Übung Imaging Science
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10171 Imaging Science (PL), schriftlich, eventuell mündlich, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben. • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	<ul style="list-style-type: none"> • 29430 Computer Vision • 55640 Correspondence Problems in Computer Vision
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 10180 Information Retrieval und Text Mining

2. Modulkürzel:	052401010	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. Sebastian Pado		
9. Dozenten:	<ul style="list-style-type: none"> • Sebastian Pado • Roman Klinger 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	052400009		
12. Lernziele:	Die Studierenden haben ein grundlegendes Verständnis der Konzepte und Algorithmen des Information Retrieval und Text Mining entwickelt.		
13. Inhalt:	<ul style="list-style-type: none"> • Textpräprozessierung • invertierte Indexe • IR-Modelle (z.B. Vektorraum-basiertes IR) • Linkanalyse • Clustering • Frage-Antwort-Systeme • Informationsextraktion • korpusbasierter Erwerb von lexikalischem und Weltwissen 		
14. Literatur:	<ul style="list-style-type: none"> • Chris Manning, Prabhakar Raghavan, Hinrich Schütze, Introduction to Information Retrieval, 2008 Cambridge University Press 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101801 Vorlesung Information Retrieval and Text Mining • 101802 Übung Information Retrieval and Text Mining 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10181 Information Retrieval und Text Mining (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0 • 10182 Information Retrieval und Text Mining - Hausübungen (USL), Sonstiges, Gewichtung: 1.0 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Maschinelle Sprachverarbeitung		

Modul: 39040 Rechnernetze

2. Modulkürzel:	051200010	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. Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> • Kurt Rothermel • Frank Dürr 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Katalog ISW 1-3 →</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051510005 Datenstrukturen und Algorithmen • Grundkenntnisse in Java 		
12. Lernziele:	<ul style="list-style-type: none"> • 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. 		
13. Inhalt:	<ul style="list-style-type: none"> • 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. 		
14. Literatur:	<ul style="list-style-type: none"> • A.S. Tanenbaum, Computer Networks, 4th Edition, 2003 • D.E. Comer, Computernetzwerke und Internets, 2000 • 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 		

	<ul style="list-style-type: none">• L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach, 1999
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 390401 VL Rechnernetze• 390402 ÜB Rechnernetze
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 39041 Rechnernetze (PL), schriftlich oder mündlich, Gewichtung: 1.0, Prüfungsdauer: 90 min schriftlich oder 30 min mündlich Exam duration: 90 min written exam or 30 min oral exam• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Verteilte Systeme

340 Katalog ISW 4

Zugeordnete Module: 31600 Machine learning for NLP

Modul: 31600 Machine learning for NLP

2. Modulkürzel:	052400616	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	2.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Sebastian Pado		
9. Dozenten:	Sebastian Pado		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Katalog ISW 4-7 → B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Katalog ISW 4 →		
11. Empfohlene Voraussetzungen:	Statistical natural language processing (recommended)		
12. Lernziele:	Students have acquired in-depth knowledge of several machine learning methods that are used in natural language processing and are familiar with the relevant literature.		
13. Inhalt:	<ul style="list-style-type: none"> - Maximum entropy models - Regression and regularized regression - Support vector machines - Sequence models - Generative models - Parameter estimation 		
14. Literatur:	Abney, Semisupervised Learning for Computational Linguistics, Chapman and Hall/CRC, 2007. Bishop, Pattern Recognition and Machine Learning, Springer, 2007.		
15. Lehrveranstaltungen und -formen:	316001 Seminar course Machine learning for NLP		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 28h Selbststudium: 60h		
17. Prüfungsnummer/n und -name:	31601 Machine learning for NLP (BSL), schriftlich, eventuell mündlich, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Maschinelle Sprachverarbeitung		

350 Wahlmodule aus Master Informatik

Zugeordnete Module:	10040	Bildsynthese
	10080	Datenbanken und Informationssysteme
	10120	Modellbildung und Simulation
	10250	Parallele Systeme
	13200	BWL III: Marketing und Einführung in die Wirtschaftsinformatik
	14380	Hardware Verification and Quality Assessment
	14480	Sichere und zuverlässige Softwaresysteme
	14740	Partielle Differentialgleichungen (Modellierung, Analysis, Simulation)
	16500	Software Engineering
	24900	Fachpraktikum Rechnergestützte Methoden der Produktentwicklung
	29330	Vertiefungslinie Datenbanken und Informationssysteme
	29340	Vertiefungslinie Intelligent Systems
	29370	Vertiefungslinie Rechnerarchitekturen und eingebettete Systeme
	29380	Vertiefungslinie Theoretische Informatik und Wissenschaftliches Rechnen
	29390	Vertiefungslinie Verteilte Systeme
	29400	Vertiefungslinie Visualisierung und Interaktive Systeme
	29410	Diskrete Optimierung
	29420	Konkrete Mathematik
	29430	Computer Vision
	29440	Geometric Modeling and Computer Animation
	29450	Graphentheorie
	29460	Kryptographische Verfahren
	29470	Machine Learning
	29480	Loose Coupling and Message Based Applications
	29500	Visual Computing
	29510	Service Computing
	29550	Algorithmische Geometrie
	29570	Computer Interface Technologien
	29580	Data Compression
	29590	Digitale Systeme
	29600	Digital System Design II
	29610	Hardware Based Fault Tolerance
	29640	Mikrocontroller
	29650	Parallele Programmierung
	29660	Programmanalysen und Compilerbau
	29670	Rapid Prototyping
	29680	Real-Time Programming
	29690	Real-Time Video Processing I
	29700	Real-Time Video Processing II
	29710	Embedded Systems Engineering
	29720	Mobile Computing
	29730	Modelling, Simulation, and Specification
	29740	Fachpraktikum Eingebettete Systeme
	29750	Fachpraktikum Rechnerarchitektur
	29760	Algorithmische Gruppentheorie
	31080	Service Engineering
	34940	Weiterführende Numerik partieller Differentialgleichungen
	36410	Requirements Engineering und Software-Architektur
	39250	Distributed Systems I
	40620	Hauptseminar INF 1
	40630	Ringvorlesung Informatik
	40680	Optimization
	41980	Grundlagen der VWL
	42010	BWL I: Produktion, Organisation, Personalführung, Strategisches Management

42220 Marketing I
42420 High Performance Computing
42460 Numerische Simulation
42480 Ausgewählte Kapitel des Wissenschaftlichen Rechnens
42520 Services and Service Composition
42810 Software-Qualitätssicherung und -Wartung
42900 Business Process Management
42910 Advanced Business Process Management
42920 Hardware-Software-Codesign
43060 Hauptseminar INF 2
45730 Distributed Systems II
45740 Rechnernetze II
45750 Fachpraktikum Verteilte Systeme
45760 Ausgewählte Kapitel der Algorithmentheorie
46430 Grundlagen der Betriebswirtschaftslehre
46440 Vertiefungslinie Programmiersprachen, Compilerbau, Programmanalysen
46450 Vertiefungslinie Architektur von Anwendungssystemen
46460 Vertiefungslinie Sprachverarbeitung
46470 Vertiefungslinie Parallele Systeme
46660 Service Management and Cloud Computing, and Evaluation
48500 Image Synthesis
48570 Practical Course Visual Computing
48580 Reinforcement Learning
48600 Robotics I
48610 Robotics II
48620 Scientific Visualization
48640 Theoretical and Methodological Foundations of Autonomous Systems
51540 Implementierung Finiter Elemente
51720 IT-Strategy
51740 Quantencomputing
55600 Advanced Information Management
55610 Information Integration
55620 Data Warehousing, Data Mining, and OLAP
55630 Information Visualization and Visual Analytics
55640 Correspondence Problems in Computer Vision
55650 Multimodal Interaction for Ubiquitous Computers
55740 Advanced Service Computing
56550 Software Verification
56680 Automaten über unendlichen Objekten
56790 Parallele Numerik
56980 Fachpraktikum Cloud Architekturen und Management
57050 Compilerbau
57680 Einführung in die Chaostheorie
58190 Entwurf und Implementierung eines Compilers
58440 Fachpraktikum: Algorithmik
60120 Interaktive Systeme
60140 Sprachbau mit Language Workbenches

Modul: 42910 Advanced Business Process Management

2. Modulkürzel:	052010007	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Frank Leymann • Dimka Karastoyanova 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	052010006 Workflow Management 1		
12. Lernziele:	<p>Am Ende der Veranstaltungen haben die Teilnehmer weiterführende Ansätze zur Modellierung von Prozessen und zur Spezifikation von Workflows verstanden. Die Rolle von Muster in der Beschreibung von Workflows ist klar geworden. Verfahren des Process Mining sind theoretisch dargestellt. Die Notwendigkeit zur P2P-Verzahnung ("Choreographien") von Prozessen und entsprechende Ansätze sind klar. Ebenso verstanden ist das darüber hinausgehende Konzept der Komponentenverdrahtung. Weitere Architekturen und Einsatzgebiete von WFMS sind verstanden.</p>		
13. Inhalt:	<p>In der Vorlesung werden fortgeschrittene Themen des Workflowmanagement vorgestellt. Aktuelle Entwicklungen aus dem Forschungsumfeld und der Industrie auf dem Gebiet werden diskutiert.</p> <p>Human Task Management Weitere Ansätze zur Prozessmodellierung (Pi-Kalkül, WSFL, XLANG,...) Muster (Kontrol-, Datenfluss, Organisatorisch) Process Monitoring Process Mining Peer-to-Peer Verzahnung von Prozessen (Choreographie, Gebrauchsanweisungen,...) Verdrahten von Komponenten (Global Models,...) Anwendungsbereiche (Manufakturing, Compliance,...) Prozessadaption und -flexibilität</p>		
14. Literatur:	W. van der Aalst, K. van Hee, Workflow Management, 2002		
15. Lehrveranstaltungen und -formen:	429101 Vorlesung mit Übungen, Workflow Management 2		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

17. Prüfungsnummer/n und -name:

- 42911 Advanced Business Process Management (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0
- V Vorleistung (USL-V), schriftlich, eventuell mündlich

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Architektur von Anwendungssystemen

Modul: 55600 Advanced Information Management

2. Modulkürzel:	051200099	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	PD Holger Schwarz		
9. Dozenten:	<ul style="list-style-type: none"> • Holger Schwarz • Bernhard Mitschang 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Lecture "Modellierung" or comparable course		
12. Lernziele:	The students learn current concepts for modeling, developing, deploying and processing database-oriented applications. This includes technologies and standards for XML processing and their integration into database systems as well as concepts and systems for content management and data management in the cloud.		
13. Inhalt:	<p>Among the topics to be discussed in this course are:</p> <ul style="list-style-type: none"> • XML and database technology (XML modeling, XML storage, XML query languages, XML processing) • NoSQL data management (Key value stores, MapReduce, triple stores, document stores, graph stores) • Content management (Enterprise content management, information retrieval, search technologies) 		
14. Literatur:	Will be announced at the beginning of the lecture.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 556001 Vorlesung Advanced Information Management • 556002 Übung Advanced Information Management 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 55601 Advanced Information Management (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0, schriftlich (90 min) oder mündlich (20 min), written (90 min) or oral (20 min) • V Vorleistung (USL-V), schriftlich, eventuell mündlich, 90 Min. 		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von: Datenbanken und Informationssysteme

Modul: 55740 Advanced Service Computing

2. Modulkürzel:	052010005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	5.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Dimka Karastoyanova		
9. Dozenten:	<ul style="list-style-type: none"> • Dimka Karastoyanova • Frank Leymann 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Service Computing, Lecture and Exercise (4 SWS) or Services and Service Composition, Lecture and Exercise (4SWS)		
12. Lernziele:	<p>This module comprises two lectures and therefore topics from two areas of advanced service computing. The focus of the Lecture Advanced Service Computing is concepts and technologies for describing and providing stateful resources as Web Services as well as the use of Semantics in Web Services and service compositions. The focus in the Lecture Services and Security is on security aspects of service-based applications.</p>		
13. Inhalt:	<p>This module comprises two lectures and therefore topics from two areas of advanced service computing.</p> <p>Based on the topics discussed in the lecture Service Computing, in the Lecture Advanced Service Computing we will focus on concepts and technologies for describing and providing stateful resources as Web Services. In this respect we will also consider Grid Services and infrastructures. In addition, the topics Semantic Web, Ontologies and Semantic Web Services will be presented in detail. Particular attention will be paid to Semantic Web Service Technologies and frameworks like OWL-S, WSMO, SAWSDL and approaches for their use in service compositions.</p> <p>The focus in the Lecture Services and Security is on security aspects of service-based applications. Foundations of Security in enterprise architectures will be presented, as well as best practices for enterprise and IT security in terms of patterns. Basic Security approaches (e.g. prevention, detection, reaction) and mechanisms (access control, authentication, identification, cryptography) will be presented in detail. We will also discuss current state of the art of Web application and Web Service security.</p>		

14. Literatur:	<ul style="list-style-type: none">• Literatur, die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben.• S. Graham, D. Davis, S. Simeonov, G. Daniels, P. Brittenham, Y. Nakamura, P. Fremantle, D. König, C., Building Web Services with Java (2nd Edition), 2005• S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005- Markus Schumacher et al.: Security Patterns: Integrating Security and Systems Engineering, Wiley Series in Software Design Patterns, 2004- Dieter Gollman: Computer Security, John Wiley & Sons; 3rd Edition, 2010
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 557401 Advanced Service Computing Lecture (Summer)• 557402 Lecture Services and Security (Winter)
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 60 Stunden Selbststudium: 120 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 55741 Advanced Service Computing (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0, Mündliche Prüfung von 30 Min• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Architektur von Anwendungssystemen

Modul: 29550 Algorithmische Geometrie

2. Modulkürzel:	050410105	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. Stefan Funke		
9. Dozenten:	Stefan Funke		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Grundkenntnisse in Datenstrukturen und Algorithmen wie sie in "Datenstrukturen und Algorithmen", "Algorithmen und Berechenbarkeit", "Algorithmik vermittelt werden.		
12. Lernziele:	Die Teilnehmer kennen die Grundbegriffe der Algorithmischen Geometrie und haben einen Überblick über die Methoden und Techniken, die in der Algorithmischen Geometrie angewandt werden.		
13. Inhalt:	Es werden die grundlegenden Techniken und Methoden der Algorithmischen Geometrie vermittelt.		
14. Literatur:	<ul style="list-style-type: none"> • Computational Geometry-Algorithms and Applications de Berg, M., Cheong, O., van Kreveld, M., Overmars, M., Springer 		
15. Lehrveranstaltungen und -formen:	295501 Vorlesung Algorithmische Geometrie		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	29551 Algorithmische Geometrie (PL), schriftlich oder mündlich, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Algorithmik		

Modul: 29760 Algorithmische Gruppentheorie

2. Modulkürzel:	050420115	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	Volker Diekert		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Elementare Gruppentheorie		
12. Lernziele:	<p>Die Studierenden beherrschen typische Denk- und Herangehensweisen aus der algorithmischen und kombinatorischen Gruppentheorie. Sie wissen, wie man diverse algorithmische Probleme in freien Gruppen mit Hilfe der Stallingsgraphen lösen kann. Sie können mit Darstellungen von Gruppen durch Erzeugende und Relationen umgehen. Sie kennen das Wortproblem und deren Lösung für gewisse Klassen von Gruppen. Sie kennen konfluente Ersetzungssysteme, HNN-Erweiterungen, amalgamierte Produkte und die Grundbegriffe der Bass-Serre-Theorie.</p>		
13. Inhalt:	<p>Bereits 1911 formulierte Max Dehn drei fundamentale algorithmische Probleme für endlich dargestellte Gruppen.</p> <p>1. Ist ein gegebenes Gruppenelement g (als Wort in Erzeugern) das Einselement in der Gruppe G? 2. Sind zwei Elemente g und h konjugiert? 3. Definieren zwei gegebene Darstellungen isomorphe Gruppen?</p> <p>Im Allgemeinen sind alle diese Fragen unentscheidbar, also kann man positive Antworten nur in Spezialfällen erhalten. Bei der Lösung des Wortproblems und bei Strukturaussagen ist vor allem die Technik der konfluenten Wortersetzungssysteme hilfreich, die auch in anderen Bereichen zum Einsatz kommen. Insgesamt lebt die Theorie von Querbezügen zu anderen Bereichen, wie Kombinatorik, Topologie, Geometrie, theoretischer Informatik. Dieses Zusammenspiel verschiedener Methoden macht die algorithmische Gruppentheorie sehr attraktiv.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Björner, Brenti: Combinatorics of Coxeter groups, Springer, 2005. • Camps, Große Rebel, Rosenberger: Einführung in die kombinatorische und geometrische Gruppentheorie, Heidemann Verlag 2008. • Lyndon, Schupp: Combinatorial Group Theory, Springer, 1977. 		

- Magnus, Karrass, Solitar: Combinatorial Group Theory, Wiley & Sons, 1966.
- Serre: Trees, Springer, 1980.
- Stillwell: Classical Topology and Combinatorial Group Theory, Springer, 1993.

15. Lehrveranstaltungen und -formen:	297601 Vorlesung mit Übung Algorithmische Gruppentheorie
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	29761 Algorithmische Gruppentheorie (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Theoretische Informatik

Modul: 45760 Ausgewählte Kapitel der Algorithmentheorie

2. Modulkürzel:	050410115	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	<ul style="list-style-type: none"> • Volker Diekert • Stefan Funke • Ulrich Hertrampf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Grundkenntnisse in Algorithmen und Komplexität		
12. Lernziele:	Die Teilnehmer lernen aktuellste Resultate aus der Algorithmentheorie kennen.		
13. Inhalt:	Es werden aktuelle Forschungsergebnisse in der Algorithmentheorie präsentiert.		
14. Literatur:	Originalartikel		
15. Lehrveranstaltungen und -formen:	457601 Vorlesung Ausgewählte Kapitel der Algorithmentheorie		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	45761 Ausgewählte Kapitel der Algorithmentheorie (PL), schriftlich, eventuell mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Formale Methoden der Informatik		

Modul: 42480 Ausgewählte Kapitel des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240030	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • 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 • Modul 42410 Grundlagen des wissenschaftlichen Rechnens 		
12. Lernziele:	Die Teilnehmer kennen ausgewählte aktuelle Forschungsthemen des wissenschaftlichen Rechnens und können mit der zugehörigen Primärliteratur arbeiten.		
13. Inhalt:	Aktuelle weiterführende Forschungsthemen des wissenschaftlichen Rechnens, wie z.B. adaptive Finite Elemente, hierarchische Basen und dünne Gitter, robuste Multilevellöser, Wavelets und schnelle Wavelettransformation, p-Version oder Spektralverfahren.		
14. Literatur:	Primärliteratur zu den behandelten Themen: <ul style="list-style-type: none"> • Bungartz/Griebel: Sparse Grids; Acta Numerica, Volume 13, p. 147-269 • Quarteroni/Valli: Numerical approximation of partial differential equations • Quarteroni: Numerical models for differential problems 		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 424801 Vorlesung Ausgewählte Kapitel des Wissenschaftlichen Rechnens• 424802 Übung Ausgewählte Kapitel des Wissenschaftlichen Rechnens
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudiumszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	42481 Ausgewählte Kapitel des Wissenschaftlichen Rechnens (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Simulation großer Systeme

Modul: 56680 Automaten über unendlichen Objekten

2. Modulkürzel:	050420230	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	Volker Diekert		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Grundkenntnisse in Mathematik und Theoretischer Informatik. (reguläre Sprachen und endliche Automaten).		
12. Lernziele:	Die Studierenden lernen die wichtigsten Grundtechniken in dem Bereich der formalen Verifikation für nicht terminierende Systeme und nebenläufige Prozess kennen. Sie lernen Denkweisen und Resultate aus verschiedenen mathematischen Disziplinen wie der Topologie, der Logik, oder der Kombinatorik kennen. Sie kennen den Begriff der MSO-Logik und ihre Entscheidbarkeit nach Büchi und Rabin.		
13. Inhalt:	<p>Die Vorlesung behandelt eine mathematischen Theorie für nicht terminierende Systeme und nebenläufige Prozess. Bei der formalen Verifikation kommen Automatenmodelle zum Einsatz, welche unendliche Objekte als Eingabe erhalten. So lassen sich viele Methoden von endlichen Wörtern auf weitere Bereiche wie unendliche Sequenzen oder Bäume ausdehnen. In diesem Sinne ist die Automatentheorie über unendlichen Objekten wesentlich reichhaltiger und spannender als über endlichen Wörtern. Die Vorlesung orientiert sich an den folgenden Themen:</p> <ul style="list-style-type: none"> • Presburger Arithmetik: Anforderungen an Automaten • Büchi Automaten und omega-reguläre Sprachen • Klarlunds Konstruktion zur Komplementierung von Büchi Automaten • Andere Akzeptanzbedingungen für omega-Automaten • Monadische Logik zweiter Stufe (MSO) • Deterministische omega-Sprachen • Topologisch definierte Sprachklassen • McNaughtons Theorem • Die Safra-Konstruktion • Algebraische Beschreibungen • Eindeutige Büchi Automaten • Logik erster Stufe und andere Fragmente von MSO • Paritätsspiele 		

- Automaten über unendlichen Bäumen
- Rabins Baumtheorem

14. Literatur:

- Volker Diekert, Manfred Kufleitner, Gerhard Rosenberger: Diskrete algebraische Methoden: Arithmetik, Kryptographie, Automaten und Gruppen. De Gruyter, Berlin 2013.
- Volker Diekert und Paul Gastin: First-order definable languages. In Jörg Flum, Erich Grädel, Thomas Wilke (eds.). Logic and Automata: History and Perspectives. Texts in Logic and Games 2, Amsterdam University Press 2008, pp. 261-306.
- Wolfgang Thomas: Automata on infinite objects. In Jan van Leeuwen (ed.). Handbook of Theoretical Computer Science, volume B: Formal Models and Semantics. Elsevier, 1990, pp. 133-192.
- Wolfgang Thomas: Languages, Automata, and Logic. In Grzegorz Rozenberg and Arto Salomaa (eds). Handbook of Formal Languages, volume 3: Beyond Words. Springer, New York, 1997, pp. 389-455.

15. Lehrveranstaltungen und -formen:

566801 Vorlesung Automaten über unendlichen Objekten

16. Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden
 Selbststudium: 138 Stunden

17. Prüfungsnummer/n und -name:

56681 Automaten über unendlichen Objekten (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 42010 BWL I: Produktion, Organisation, Personalführung, Strategisches Management

2. Modulkürzel:	100140120	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. Rudolf Large		
9. Dozenten:	<ul style="list-style-type: none"> • Rudolf Large • Gordon Müller-Seitz • Michael-Jörg Oesterle 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:	Grundlagen der BWL		
12. Lernziele:	<p>Aus den aufgeführten drei Lehrveranstaltungen sind für das Bestehen des Moduls zwei Lehrveranstaltungen auszuwählen.</p> <p>Veranstaltung "Produktionsmanagement":</p> <p>Die Studierenden sind am Ende der Veranstaltung in der Lage,</p> <ul style="list-style-type: none"> • Produktionssysteme mit Hilfe von Produktions- und Kostenfunktionen abzubilden, • produktionswirtschaftliche Fragestellungen in Planungsmodellen abzubilden, • grundlegende Planungsmethoden der Produktion anzuwenden. <p>Veranstaltung "Organisation und Personalführung":</p> <p>Die Studierenden verfügen über Grundkenntnisse zum Aufbau und zum Prozess der Gestaltung von Produktionssystemen für Sach- und Dienstleistungen sowie von Führungssystemen (Kenntnisse der zentralen Führungsaufgaben auf den Gebieten der Organisationsgestaltung, Personalentwicklung, Personalbeschaffung, Personalbindung und Personalfreisetzung und des Aufbaus von Anreizsystemen). Die Studierenden sind in der Lage, ausgewählte Führungsmethoden anzuwenden.</p> <p>Veranstaltung "Strategisches Management"</p> <p>Die Studierenden sollen zunächst Bedeutung und Notwendigkeit des strategischen Managements, aber auch dessen Grenzen erkennen können, darüber hinaus sollen sie in der Lage sein, vor dem Hintergrund der Entwicklung des strategischen Denkens in der Betriebswirtschaftslehre und in der Unternehmenspraxis theoretisch fundiert Konzepte und Instrumente des strategischen Managements kritisch zu analysieren sowie in ihrem Anwendungsbezug beurteilen zu können.</p>		

13. Inhalt:

Aus den aufgeführten drei Lehrveranstaltungen sind für das Bestehen des Moduls zwei Lehrveranstaltungen auszuwählen.

Veranstaltung "Produktionsmanagement":

Gegenstand der Vorlesung sind zunächst die Grundlagen der Produktions- und Kostentheorie. Darauf baut die Behandlung der grundlegenden Teilaufgaben der Produktionsplanung und -steuerung auf: Produktionsprogrammplanung, Materialbedarfsplanung und Losgrößenrechnung, Durchlaufplanung und Fertigungssteuerung. In der Übung werden die zugehörigen Planungsmethoden der Produktion angewendet.

Veranstaltung "Organisation und Personalführung":

Funktionelle, institutionelle, personelle und instrumentelle Zugänge zu Führungssystemen; Führungsstile und Führungsmodelle; Dezentralisierung der Personalführung; interaktionelle und infrastrukturelle Führung. Grundlagen der Qualifizierung, Rekrutierung und Motivierung (Aufbau von Anreizsystemen); Eingliederung und Aufgliederung der Organisationsgestaltung; Organisationsstrukturen; Organisationsprozesse; Projektorganisation; Center-Konzepte; Matrixorganisation; Koordinationsorgane; Kontextfaktoren: Strategie, Personal und Technologie; Organisationsstrukturen für das internationale und das Produktgeschäft.

Veranstaltung "Strategisches Management":

Überblick über die Entwicklung des Strategischen Managements in Theorie und Praxis; Theoretische Ansätze des Strategischen Managements; Akteure und Inhalte des Strategischen Managements; Prozess, Methoden und Techniken der Strategieformulierung; Ansätze zur Implementierung von Strategien; Fit- bzw. stimmigkeitsbezogene Ansätze im Strategischen Management; Normative Konzepte der strategischen Unternehmensgestaltung; Strategien international tätiger Unternehmen.

14. Literatur:

- Skript Produktionsmanagement
- Skript Organisation und Personalführung
- Skript Strategisches Management

Veranstaltung "Produktionsmanagement":

- Large, Rudolf: Betriebswirtschaftliche Logistik. Band 1: Logistikfunktionen. Neueste Auflage.
- Bloech, Jürgen et al.: Einführung in die Produktion. Neueste Auflage.
- Günther, Hans-Otto/ Tempelmeier, Horst: Produktion und Logistik. Neueste Auflage.
- Tempelmeier, Horst: Material-Logistik. Modelle und Algorithmen für die Produktionsplanung und -steuerung in Advanced Planning-Systemen. Neueste Auflage.

Veranstaltung "Strategisches Management":

- Bamberger, I., Wrona, T.: Strategische Unternehmensführung. Neueste Auflage.
- De Witt, B., Meyer, R.: Strategy - Process, content, context - an international perspective. Neueste Auflage.

- Johnson, G., Scholes, K., Whittington, R.: Strategisches Management - Eine Einführung, Analyse, Entscheidung und Umsetzung. Neueste Auflage.
- Volberda, H. W. et al.: Strategic Management - Competitiveness and Globalization. Neueste Auflage.
- Welge, M. K., Al-Laham, A.: Strategisches Management - Grundlagen, Prozesse, Implementierung. Neueste Auflage.

15. Lehrveranstaltungen und -formen:

- 420101 Vorlesung Produktmanagement
- 420102 Übung Produktmanagement
- 420103 Vorlesung Organisation und Personalführung
- 420104 Übung Organisation und Personalführung
- 420105 Vorlesung Strategisches Management
- 420106 Übung Strategisches Management

16. Abschätzung Arbeitsaufwand:

Vorlesung (jeweils)

Präsenzzeit: 28 h

Selbststudiumszeit: 62 h

Übung (jeweils)

Präsenzzeit: 14 h

Selbststudiumszeit: 31 h

Gesamtstundenzahl: 270 h

17. Prüfungsnummer/n und -name:

42011 BWL I: Produktion, Organisation, Personalführung, Strategisches Management (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0, Aus den aufgeführten drei Lehrveranstaltungen sind zwei Lehrveranstaltungen auszuwählen. Für das Bestehen des Moduls ist die Prüfung über die Inhalte der beiden ausgewählten Lehrveranstaltungen abzulegen.

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

ABWL und Dienstleistungsmanagement, insbesondere Unternehmenslogistik

Modul: 13200 BWL III: Marketing und Einführung in die Wirtschaftsinformatik

2. Modulkürzel:	100160001	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. Hans-Georg Kemper		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Georg Kemper • Georg Herzwurm • Torsten Bornemann 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:	Grundlagen der Betriebswirtschaftslehre		
12. Lernziele:	<p>Marketing: Die Studierenden haben einen Überblick über das gesamte Stoffgebiet des Fachs Marketing und verfügen über grundlegende Kenntnisse.</p> <p>Einführung in die Wirtschaftsinformatik: Die Studierenden können die betriebswirtschaftliche Relevanz von Informationssystemen einschätzen. Sie verfügen über Kenntnisse zu Formen und Komponenten von Informationssystemen sowie zu den Gegenständen und Inhalten der Wissenschaft Wirtschaftsinformatik.</p>		
13. Inhalt:	<p>Marketing: Allgemeine Grundlagen; Theoretische Perspektive: Das Verhalten der Kunden; Informationsbezogene Perspektive: Marktforschung; Strategische Perspektive: Strategisches Marketing; Instrumentelle Perspektive: Produktpolitik, Preispolitik, Kommunikationspolitik, Distributions- und Vertriebspolitik; Institutionelle Perspektive: Dienstleistungsmarketing, Business-to-Business-Marketing, Internationales Marketing.</p> <p>EiW: Im Zuge der zunehmenden Durchdringung betrieblicher Prozesse mit Informationstechnologie (IT) rücken Fragen einer zielgerichteten Gestaltung und Nutzung von IT-basierten Lösungen immer mehr in den Mittelpunkt betriebswirtschaftlichen Handelns. Entwicklung und Anwendung von Informations- und Kommunikationssystemen (IuK-Systeme) als sozio-technische Lösungen in Wirtschaft und Verwaltung sind Gegenstände der Disziplin "Wirtschaftsinformatik". Die Veranstaltung stellt die Wirtschaftsinformatik vor und gibt einen ein Überblick über die von ihr adressierten Themenkomplexe sowie über grundlegende Theorien, Methoden und Konzepte des Fachs.</p>		
14. Literatur:	<p>Marketing:</p> <ul style="list-style-type: none"> • Vorlesungsskript und Übungsunterlagen 		

- Homburg, Ch. (2012), Grundlagen des Marketingmanagements, 3. Auflage, Wiesbaden.
- Homburg, Ch. (2012), Marketingmanagement, 4. Auflage, Wiesbaden. (vertiefend)

Einführung in die Wirtschaftsinformatik:

- Laudon, K. C., Laudon, J. P., Schoder, D.: Wirtschaftsinformatik, eine Einführung, aktuelle Auflage
- Stahlknecht, P., Hasenkamp, U., Einführung in die Wirtschaftsinformatik, aktuelle Auflage
- Hansen, H. R., Neumann, G.: Wirtschaftsinformatik 1, aktuelle Auflage
- Skript

15. Lehrveranstaltungen und -formen:

- 132001 Vorlesung Marketing
- 132002 Übung Marketing
- 132003 Vorlesung Einführung in die Wirtschaftsinformatik
- 132004 Übung Einführung in die Wirtschaftsinformatik

16. Abschätzung Arbeitsaufwand:

Präsenzzeit:	63 h
Selbststudiumszeit / Nacharbeitszeit:	207 h
Gesamt:	270 h

17. Prüfungsnummer/n und -name: 13201 BWL III: Marketing und Einführung in die Wirtschaftsinformatik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Betriebswirtschaftliches Institut

Modul: 10040 Bildsynthese

2. Modulkürzel:	051900012	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. Thomas Ertl		
9. Dozenten:	<ul style="list-style-type: none"> • Martin Fuchs • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051900002 Computergraphik 		
12. Lernziele:	<p>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. Darüber hinaus kennen sie interaktive Verfahren, die unter Ausnutzung programmierbarer Grafik-Hardware realistische Beleuchtungseffekte in Echtzeit approximieren können, sowie bildbasierte Ansätze, die ohne geometrische Daten realistische Darstellungen erzeugen. Bild-basierte Verfahren verzichten auf eine geometrische Repräsentation der Szene und erzeugen neue Ansichten aus anderen aufgenommenen Bildern.</p>		
13. Inhalt:	<p>In dieser Vorlesung werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none"> • 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 • Bild-basiertes Rendering 		

14. Literatur:	<ul style="list-style-type: none"> • Andrew S. Glassner, Principles of Digital Image Synthesis, 1995 • 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 • Matt Pharr, Greg Humphreys, Physically Based Rendering: From Theory To Implementation, Morgan Kaufmann; Auflage: 2nd revised edition. (26. August 2010) • Peter Shirley et al, Fundamentals of Computer Graphics, Third Edition, A.K. Peters, July 2009
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 100401 Vorlesung Bildsynthese • 100402 Übung Bildsynthese
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10041 Bildsynthese (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich, Übungsschein.
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 42900 Business Process Management

2. Modulkürzel:	052010006	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Frank Leymann • Dimka Karastoyanova 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	611 Grundlagen der Architektur von Anwendungssystemen, Vorlesung mit Übung, 4,0 SWS		
12. Lernziele:	<p>The course has the objective to provide knowledge about the essential modelling constructs for workflows and their mapping to corresponding workflow languages. In addition, the life cycle of Workflow-based applications will be presented in detail and connected to the Architecture of Workflow Management Systems, which will also be presented. Moreover, the goal is to enable students to use workflow languages (in particular BPEL) in practice. In this respects students will also understand the fundamental approach process graphs, which is applied in workflow languages. Of great importance are , mechanisms for fault handling and exception handling - these will be explained in detail and students will be able to apply them.</p>		
13. Inhalt:	<p>Workflows are IT realisations of business processes and are also considered an approach of significant importance for composition of applications. This course will introduce the foundations of this area, also known as Business Process Management BPM).</p> <ol style="list-style-type: none"> 1. Historical Development of the Workflow Technology 2. Business Re-engineering (BPM Lifecycle, Tools,...) 3. Architecture of WFMS (Navigator, Executor, Worklist Manager,...) 4. Flow Languages (FDL, BPEL) 5. Process Model Graph (mathematical meta-model: syntax, operational semantics) 6. Advanced functions (sub-processes, event handling, instance modifications, adaptation) 7. Two-level programming paradigm 8. Transactional support in workflows 		
14. Literatur:	<ul style="list-style-type: none"> • F. Leymann, D. Roller, Production Workflow, 2000 • W. van der Aalst, K. van Hee, Workflow Management, 2002 		

15. Lehrveranstaltungen und -formen:	429001 Vorlesung mit Übungen, Workflow Management 1
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 42901 Business Process Management (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0, Wenn Prüfung in diesem Fach, dann K E I N E Prüfung in Services and Service Composition (auch nicht innerhalb der VTL) möglich!• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Architektur von Anwendungssystemen

Modul: 57050 Compilerbau

2. Modulkürzel:	051010201	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. Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>Grundkenntnisse aus den Einführungsvorlesungen des Informatikgrundstudiums, sowie einige Erfahrungen mit Programmierung. Vorkenntnisse über formale Sprachen sind vorteilhaft, aber nicht zwingend.</p>		
12. Lernziele:	<p>Die Studierenden haben die Grundkenntnisse erlangt, die zur effizienten Verwendung von Lexer- und Parser-Generatoren zur Analyse von Eingabetexten nötig sind. Sie verstehen die grundlegende Funktionsweise mehrerer Parse-Verfahren und kennen deren grammatikalischen Einschränkungen. Sie kennen elementare Verfahren semantischer Analysen und sind in der Lage, einfache semantische Prüfungen zu verfassen. Sie haben gelernt, die Fehlermeldungen aus Parser-Generatoren, Compilern oder Interpretern richtig einzuordnen. Ferner haben sie durch Betrachtung der Implementierungsmodelle typischer Programmiersprachenkonstrukte Verständnis für das Ausführungsverhalten erlangt. Sie kennen elementare Begriffe der Codegenerierung und die Eigenschaften von typischen Zwischencodedarstellungen in Compilern.</p>		
13. Inhalt:	<p>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. Attributgrammatiken. Zwischencodeerzeugung. Realisierung einiger Aspekte der Laufzeitsemantik prozeduraler Programmiersprachen. Einfache Codegenerierung.</p>		
14. Literatur:	<ul style="list-style-type: none"> • A. V. Aho, M.S. Lam, R. Sethi, J. D. Ullman: Compilers - Principles, Techniques, and Tools, Addison Wesley Verlag (2007) 		

	<ul style="list-style-type: none">• Niklaus Wirth: Compilerbau: Eine Einführung, Teubner Verlag (1986)• Wilhelm, Maurer: Übersetzerbau, 2. Auflage, Springer Verlag (1997)• Andrew W. Appel: Modern Compiler Implementation In Java, Cambridge University Press (2002)• Uwe Kastens: Übersetzerbau, Oldenbourg Verlag (1990)
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 570501 Vorlesung Compilerbau• 570502 Übung Compilerbau
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 h Selbststudiumszeit / Nacharbeitszeit: 138 h Gesamt: 180 h
17. Prüfungsnummer/n und -name:	57051 Compilerbau (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	29660 Programmanalysen und Compilerbau
19. Medienform:	
20. Angeboten von:	Programmiersprachen und ihre Übersetzer

Modul: 29570 Computer Interface Technologien

2. Modulkürzel:	051230105	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Kenntnisse in mindestens einem Fach der Technischen Informatik oder ähnlichen Fächern und Erfahrungen in mindestens einer Programmiersprache.		
12. Lernziele:	Die Studierenden haben die Funktionsweise und den Aufbau von Coputer Interfaces verstanden. Sie beherrschen verschiedene Interface-Konzepte und kennen die Eigenschaften der Datenströme wie Latenzzeit, tatsächliche Durchsatzrate, Echtzeitfähigkeit, Umgang mit Übertragungsfehlern, etc.		
13. Inhalt:	<ul style="list-style-type: none"> • Grundlagen - Computer Interfaces • Computer Interfaces und OSI-Modelle • Bus- und Netz-Topologien • Line und Error Codes • Protokolle • Treiber • Compliance Tests • Standardization Groups: USB, PCI, etc. 		
14. Literatur:	<ul style="list-style-type: none"> • Patterson, David A.; Hennessey, John L., Computer Organization and Design - The Hardware / Software Interface, 2008 <p>More literature is named in the lecture.</p>		
15. Lehrveranstaltungen und -formen:	295701 Vorlesung mit Übung Computer Interface Technologien		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	29571 Computer Interface Technologien (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0, Schriftliche Prüfung von 120 Minuten oder mündliche Prüfung von 30 Min.		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 29430 Computer Vision

2. Modulkürzel:	051900215	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10190 Mathematik für Informatiker und Softwaretechniker • Modul 10170 Imaging Science 		
12. Lernziele:	<p>Der Student / die Studentin beherrscht die Grundlagen der Merkmalsextraktion und -repräsentation, des 3-D Maschinensehens, der Bildsegmentierung sowie der Mustererkennung. Er/sie kann Probleme aus dem Fachgebiet einordnen und diese selbständig mit den erlernten Algorithmen und Verfahren lösen.</p> <p>The student knows the basics of feature extraction and representation, 3-D computer vision, image segmentation and pattern recognition. He/ she can solve problems of the field using the methods discussed in the course.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • Lineare Diffusion, Skalenräume • Bildpyramiden, Kanten und Eckendetektion • Hough-Transformation, Invarianten • Texturanalyse • Scale Invariant Feature Transform (SIFT) • Bildfolgenanalyse: lokale Verfahren • Bewegungsmodelle, Objektverfolgung, Feature Matching • Bildfolgenanalyse: globale Verfahren • Kamerageometrie, Epipolargeometrie • Stereo Matching und 3-D Rekonstruktion • Shape-from-Shading • Isotrope und anisotrope nichtlineare Diffusion • Segmentierung mit globalen Verfahren • Kontinuierliche Morphologie, Schockfilter • Mean Curvature Motion • Self-Snakes, Aktive Konturen • Bayes'sche Entscheidungstheorie der Mustererkennung • Klassifikation mit parametrischen Verfahren, Dichteschätzung • Klassifikation mit nicht-parametrischen Verfahren 		

- Dimensionsreduktion
- Linear Diffusion, Scale Space
- Image Pyramids, Edges and Corners
- Hough Transform, Invariants
- Texture Analysis
- Scale Invariant Feature Transform
- Image Sequence Analysis: Local Methods
- Motion Models, Tracking, Feature Matching
- Image Sequence Analysis: Variational Methods
- Camera Geometry, Epipolar Geometry
- Stereo Matching and 3-D Reconstruction
- Shape-from-Shading
- Isotropic and Anisotropic Nonlinear Diffusion
- Segmentation with Global Methods
- Continuous Scaled Morphology, Shock Filters
- Mean Curvature Motion
- Self-Snakes, Active Contours
- Bayes Decision Theory for Pattern Recognition
- Classification with Parametric Techniques, Density Estimation
- Classification with Non-Parametric Techniques
- Dimensionality Reduction

14. Literatur:	<ul style="list-style-type: none"> • Forsyth, David and Ponce, Jean, Computer Vision. A Modern Approach.: A Modern Approach Computer Vision. A Modern Approach, 2003 • Bigun, J.: Vision with Direction, 2006 • L. G. Shapiro, G. C. Stockman, Computer Vision, 2001 • O. Faugeras, Q.-T. Luong: The Geometry of Multiple Images, 2001.
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 294301 Vorlesung Computer Vision • 294302 Übung Computer Vision
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden</p> <p>Gesamt: 180 Stunden</p>
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29431 Computer Vision (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	55640 Correspondence Problems in Computer Vision
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 55640 Correspondence Problems in Computer Vision

2. Modulkürzel:	051900211	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 080300100 Mathematik für Informatiker und Softwaretechniker • Modul 050700005 Imaging Science • Modul 051900215 Computer Vision 		
12. Lernziele:	<p>Der Student kann Korrespondenzprobleme im Computer-Vision-Bereich selbständig einordnen, Lösungsstrategien mathematisch modellieren und diese dann geeignet algorithmisch umsetzen.</p> <p>The student has knowledge on the different correspondence problems in computer vision, is able to develop mathematical models for solution strategies and implement the corresponding algorithms in an appropriate way.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • Basisverfahren: Block Matching, Detektion von Verdeckungen, Merkmalsfindung, Feature Matching • Optischer Fluss: Lokale und Globale differentiale Verfahren, Parametrisierungsmodelle, Konstanzannahmen, Daten- und Glattheitsterme, Numerik, Große Verschiebungen, Hochgenaue Verfahren • Stereorekonstruktion: Projektive Geometrie, Epipolargeometrie, Schätzung der Fundamentalmatrix • Szenenfluss: Gemeinsame Schätzung von Struktur, Bewegung und Geometrie • Medizinische Bildregistrierung: Mutual Information, Elastische und krümmungsbasierte Regularisierung, Landmarks • Particle Image Velocimetry: Div-Curl-Regularisierung, Inkompressibler Navier Stokes Prior • Basic Approaches: Block Matching, Occlusion Detection, Interest Points, Feature Matching • Optic Flow: Local and Global Differential Methods, Parametrisation Models, Constancy Assumptions, Data and Smoothness Terms, Numerics, Large Displacements, High Accuracy Methods • Sterep Matching: Projective Geometry, Epipolar Geometry, Estimation of the Fundamental Matrix 		

	<ul style="list-style-type: none"> • Scene Flow: Joint Estimation of Structure, Motion, and Geometry • Medical Image Registration: Mutual Informaion, Elastic and Curvature-Based Regularisation, Landmarks • Particle Image Velocimetry: Div-Curl-Regularisation, Incompressible Navier Stokes Prior
14. Literatur:	<ul style="list-style-type: none"> • O. Faugeras, Q.-T. Luong: The Geometry of Multiple Images, 2001. • J. Modersitzki: Numerical Methods for Image Registration, 2003. • A. Bruhn: Variational Optic Flow Computation: Accurate Modeling and Efficient Numerics, Ph.D. Thesis, 2006.
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 556401 Vorlesung Correspondence Problems in Computer Vision • 556402 Übung Correspondence Problems in Computer Vision
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 55641 Correspondence Problems in Computer Vision (PL), schriftlich, eventuell mündlich, 120 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 29580 Data Compression

2. Modulkürzel:	051230110	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	This course requires basic knowledge in mathematics.		
12. Lernziele:	The students learn the concepts of data compression and acquire an understanding of different algorithms for data compression. Furthermore they will be able to implement and further develop the algorithms discussed in the course.		
13. Inhalt:	<ul style="list-style-type: none"> • Shannon Entropy • Huffman coding • Universal codes • Arithmetic coding • Lossy and Lossless compression • Image data compression • Dictionary based compression 		
14. Literatur:	<ul style="list-style-type: none"> • Khalid Sayood, Introduction to Data Compression, 2005 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	295801 Vorlesung mit Übung Datenkompression		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	29581 Data Compression (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0, written 90 Min. or oral 30 Min.		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 55620 Data Warehousing, Data Mining, and OLAP

2. Modulkürzel:	051210105	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Bernhard Mitschang		
9. Dozenten:	<ul style="list-style-type: none"> • Bernhard Mitschang • Holger Schwarz 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Lecture "Modellierung" or comparable course		
12. Lernziele:	<p>After attending this lecture, students understand the challenges behind the integration of heterogeneous data sources in consolidated warehouses and the provisioning of analytical services. They know the typical data warehouse architecture as well as current trends, e.g., real-time data warehousing. Further topics are the structure of a data warehouse and the main processes for building data warehouses (extraction, transformation, load). A special focus is on technologies to analyze data warehouse data, e.g., reporting, online analytic processing and data mining, and their role as part of analytical services.</p>		
13. Inhalt:	<p>Among the topics to be discussed in this course are:</p> <ul style="list-style-type: none"> - Introduction to data warehousing - Data warehouse architecture - Data warehouse design - Extraction, transformation, load - ETL as a service - Introduction to analytics and analytic services - Real-time reporting - Online analytic processing - Data mining 		
14. Literatur:	<ul style="list-style-type: none"> • A. Kemper, A. Eickler, Datenbanksysteme - Eine Einführung, 2004 • H. Garcia-Molina, J. D. Ullman, J. Widom, Database Systems. The Complete Book, 2003 <p>Will be announced at the beginning of the lecture</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 556201 Vorlesung Data Warehousing, Data Mining und OLAP-Technologien • 556202 Übung Data Warehousing, Data Mining und OLAP-Technologien 		

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:	<ul style="list-style-type: none"> • Bernhard Mitschang • Holger Schwarz 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>LAGymPO Informatik, PO 2010, 5. Semester → Ergänzendes Modul</p> <p>LAGymPO Informatik, PO 2010, 5. Semester → Wahlmodule</p> <p>KLAGymPO Informatik, PO 2010, 5. Semester → Wahlmodule</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Vorlesung Modellierung oder Gleichwertiges		
12. Lernziele:	Die Studierenden haben die erforderlichen Kenntnisse für Datenbankprogrammierer in angemessenem Umfang erworben.		
13. Inhalt:	<p>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.</p> <p>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.</p>		

14. Literatur:	<ul style="list-style-type: none">• 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:	<ul style="list-style-type: none">• 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: 29600 Digital System Design II

2. Modulkürzel:	051230122	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	This lectures requires the knowledge of "System Design I". Alternatively, knowledge of "Technische Informatik" is sufficient to follow the course.		
12. Lernziele:	The students will learn to build and implement a complex digital system by using digitals components on a circuit board, and will acquire an in-depth knowledge for implementing complex digital systems using FPGA's.		
13. Inhalt:	<ul style="list-style-type: none"> • Presentation of a case study of a digital system • Simulatable specification of the system • Architecture for Implementation using FPGAs • Design and design tools for board integration • Implementation of a digital system • Verification of a digital system 		
14. Literatur:	<ul style="list-style-type: none"> • Kou-Chuan Chang, K. C. Chang, Digital Systems Design with VHDL and Synthesis: An Integrated Approach, 1999 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	296001 Vorlesung mit Übung Digital System Design II		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	29601 Digital System Design II (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29590 Digitale Systeme

2. Modulkürzel:	051230120	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Kenntnisse in einem Fach aus der Technischen Informatik oder einem ähnlichen Gebiet.		
12. Lernziele:	Die Studierende beherrschen den Entwurf Digitaler Systeme durch die Integration von digitalen Komponenten auf einem Board und die Realisierung von digitaler Komponenten mittels FPGAs.		
13. Inhalt:	<ul style="list-style-type: none"> • Praktische Einführung in den System-Entwurf mit digitalen Komponenten wie Schnittstellenbausteinen zur Kommunikation, FPGAs, Prozessoren, intelligenten Sensoren etc. • Einführung und Verwendung der Hardware-Beschreibungssprache VHDL zum Entwurf Digitaler Systeme • Digitale Systeme und Board-Integration von digitalen Komponenten • Aufbau von Computer-Boards u. Gbit/s-Interconnects • Entwurf auf höheren Abstraktionsebenen zur schnellen Entwicklung von Prototypen 		
14. Literatur:	<ul style="list-style-type: none"> • Kou-Chuan Chang, K.C. Chang, Digital Systems Design with VHDL and Synthesis: An Integrated Approach, 1999 <p>More literature is named in the lecture.</p>		
15. Lehrveranstaltungen und -formen:	295901 Vorlesung mit Übung Digital System Design I		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29591 Digitale Systeme (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0, Schriftliche Prüfung von 120 Min. oder mündliche Prüfung von 30 Min. • V Vorleistung (USL-V), schriftlich, eventuell mündlich 		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Modul: 29410 Diskrete Optimierung

2. Modulkürzel:	050410110	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Stefan Funke		
9. Dozenten:	Stefan Funke		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF → M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->TMG-INF →		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	The participants get to know the basic techniques in discrete optimization and have a good overview of the standard methods to be able to deal with new problems instances.		
13. Inhalt:	We teach basic techniques of discrete optimization like (integer) linear programming, approximation algorithms and network flow algorithms.		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	294101 Vorlesung Diskrete Optimierung		
16. Abschätzung Arbeitsaufwand:	in class: 42 h at home: 138 h		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29411 Diskrete Optimierung (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftliche Prüfung, 120 Min. 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Algorithmik		

Modul: 39250 Distributed Systems I

2. Modulkürzel:	051200015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Programmierung und Software-Entwicklung Datenstrukturen und Algorithmen Systemkonzepte und -Programmierung		
12. Lernziele:	The Students will gain an understanding of the basic characteristics, concepts and methods of distributed systems. Furthermore, the ability to analyze existing distributed applications and platforms with regard to its specific properties will be obtained. The implementation of distributed applications as well as system platforms based on the shown methods of that course is another objective. Due to the knowledge provided in that course, the students will be able to communicate with other experts of other professional disciplines, about topics in the field of distributed systems.		
13. Inhalt:	<ol style="list-style-type: none"> 1. Introduction to distributed systems 2. System models 3. Communication: Messages, Remote Procedure Call (RPC), Remote Method Invocation RMI 4. Naming: Generating and Resolution 5. Time Management and clocks in distributed Systems: Applications, logical clocks, physical clocks, synchronization of clocks 6. Global state: concepts, snapshot algorithms, distributed Debugging 7. Transaction management: Serializability, barrier methods, 2-phase-commit-protocols 8. Data replication: primary copy, consensus-protocols and other algorithms 		

	9. Safety/Security: Methods for confidentiality, integrity, authentication and authorization 10. Multicast-algorithms: processing model, broadcast-semantics and algorithms
14. Literatur:	Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 392501 Vorlesung Verteilte Systeme• 392502 Übungen Verteilte Systeme
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 h Selbststudiumszeit / Nachbearbeitungszeit: 138 h Gesamt: 180 h
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 39251 Distributed Systems I (PL), schriftlich oder mündlich, Gewichtung: 1.0, Prüfungsdauer: 90 min schriftlich oder 30 min mündlich Exam duration: 90 min written exam or 30 min oral exam• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Verteilte Systeme

Modul: 45730 Distributed Systems II

2. Modulkürzel:	051200169	5. Moduldauer:	2 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	The Lecture requires basic knowledge from the course Distributed Systems I		
12. Lernziele:	<p>In this lecture, the aquired knowledge from the previous lecture Distributed Systems I is depend. The student will gain information about further practice-oriented problems and will implement protocols to solve those problems. The student will be capable to analyze distributed systems in terms of these problems, design, apply and develop protocols for specific applications.</p>		
13. Inhalt:	<ol style="list-style-type: none"> 1. Group communication 2. Consensus 3. Fault tolerant services 4. Wave algorithms 5. Termination 6. Garbage collection 7. Election 8. Deadlocks 9. Organisational & Introduction 		
14. Literatur:	<ul style="list-style-type: none"> • J.L. Welch, H. Attiya, Distributed Computing: Fundamentals, Simulations and Advanced Topics, 1997 <p>The event is based on a collection of scientific papers, which will be announced in the lecture.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 457301 Vorlesung Verteilte Algorithmen • 457302 Vorlesung Asynchronous Middleware Systems 		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden</p> <p>Gesamt: 180 Stunden</p>		

17. Prüfungsnummer/n und -name: 45731 Distributed Systems II (PL), schriftlich oder mündlich,
Gewichtung: 1.0, Prüfungsdauer: 90 min schriftlich oder 30
min mündlichExam duration: 90 min wriitten exam or 30 min
oral exam

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Verteilte Systeme

Modul: 57680 Einführung in die Chaostheorie

2. Modulkürzel:	074810350	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. Frank Allgöwer		
9. Dozenten:	Viktor Avrutin		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009 → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012 → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012 → Vorgezogene Master-Module M.Sc. Informatik, PO 2012 → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012 → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>Die Teilnehmer lernen die Grundbegriffe der Theorie der nichtlinearen dynamischen Systeme bzw. der Chaostheorie kennen. Die Studierenden verstehen solche Begriffe wie zeit-kontinuierliche und zeit-diskrete Modellierung, transiente und asymptotische Dynamik, Attraktoren, Stabilität, Bifurkationen, Bifurkationsszenarien, Deterministisches Chaos, "Wege ins Chaos". Sie können verschiedene Typen von lokalen und globalen Bifurkationen erkennen und kennen auch die Bedingungen, die zu diesen Bifurkationen führen. Darüber hinaus lernen die Studierenden die typischen quantitativen Maße kennen, die bei der praktischen Untersuchung des Verhaltens angewendet werden. Dazu zählen in erster Linie Lyapunov Exponenten, fraktale Dimensionen und Entropien. Ein wesentlicher Teil der Vorlesung ist einem modernen Kapitel der Nichtlinearen Dynamik gewidmet, nämlich der Theorie der stückweise-glatte Systeme. Die Studierenden lernen die für diese Systeme charakteristischen Phänomene (border-collision bifurcations, period-adding) kennen, sowie Konzepte der Symbolischen Dynamik und die typischen Anwendungen aus dem technischen Bereich (impacting systems, switching circuits). Abschließend wird in der Vorlesung der Zusammenhang zwischen dynamischen Systemen und Fraktalen gezeigt. Die Studierenden verstehen darauf die Bedeutung der Standard-Beispiele aus diesem Gebiet (Cantor-Mengen, Julia-Mengen, Mandelbrot-Mengen). Ein besonderer Wert wird in dieser Lehrveranstaltung darauf gelegt, dass die Teilnehmer eigene praktische Erfahrungen im Umgang mit dynamischen Systemen (am Beispiel von niedrig-dimensionalen zeit-diskreten Abbildungen) sammeln. Zu diesem Zweck bietet die Vorlesung den Studierenden die Möglichkeit, viel zu experimentieren.</p>		
13. Inhalt:	1. Problemstellungen und Grundbegriffe		

2. Qualitative Analyse: Attraktoren (periodische, aperiodische, chaotische Trajektorien), Bifurkationen (lokale und globale Bifurkationen, Bifurkationen in stückweise-glaten Systemen); Bifurkations-szenarien (in glatten und stückweise-glaten Systemen)

3. Quantitative Analyse: Lyapunov Exponenten, fraktale Dimensionen, weitere Maße. Symbolische Dynamik

4. Fraktale

14. Literatur: John Argyris, Gunter Faust, Maria Haase, Rudolf Friedrich , Die Erforschung des Chaos: Eine Einführung in die Theorie nichtlinearer Systeme (Springer, 2010)

Skript

15. Lehrveranstaltungen und -formen: 576801 Vorlesung Einführung in die Chaostheorie

16. Abschätzung Arbeitsaufwand: Präsenzzeit: 42;
Selbststudium: 138

17. Prüfungsnummer/n und -name: 57681 Einführung in die Chaostheorie (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 29710 Embedded Systems Engineering

2. Modulkürzel:	051711027	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	keine		
12. Lernziele:	Master-level understanding of the design methodology and advanced design techniques for constructing and analyzing embedded hardware / software systems.		
13. Inhalt:	<ol style="list-style-type: none"> 1. Introduction to embedded systems and their design constraints 2. Synthesis models and algorithms 3. System level synthesis 4. High level synthesis 5. Pipelined data path and controller design 6. Software task scheduling and schedulability analysis 7. Static and dynamic methods for scheduling and priority assignment 8. Communication architectures for embedded systems 		
14. Literatur:	<p>Skript „Embedded Systems Engineering“</p> <p>G. Buttazzo: Hard Real Time Computing Systems. 2nd edition, Springer, 2005</p> <p>P. Eles, K. Kuchcinski, Z. Peng: System Synthesis with VHDL. Kluwer Academic Publishers, 1998.</p> <p>P. Marwedel: Embedded Systems Design. Springer, 2006</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 297101 Vorlesung Embedded Systems Engineering • 297102 Übung Embedded Systems Engineering 		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden</p> <p>Selbststudium: 138 Stunden</p> <p>Summe: 180 Stunden</p>		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29711 Embedded Systems Engineering (Klausur) (PL), schriftlich, eventuell mündlich, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich, Als Zulassungsvoraussetzung zur Klausur ist folgende Vorleistung 		

zu erbringen: Teilnahme an den Übungen, Präsentation der Lösung wenigstens einer Aufgabe.

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Eingebettete Systeme (Embedded Systems Engineering)

Modul: 58190 Entwurf und Implementierung eines Compilers

2. Modulkürzel:	05151313	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Erhard Plödereder		
9. Dozenten:	<ul style="list-style-type: none"> • Erhard Plödereder • Timm Felden 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>Modul Compilerbau ist notwendige Voraussetzung, Java-Kenntnisse werden erwartet.</p> <p>Die Teilnehmerzahl in diesem Modul ist auf maximal 15 beschränkt.</p>		
12. Lernziele:	<p>Die Studierenden haben praktische Erfahrungen mit der Konstruktion eines Compilers und der Umsetzung von Konzepten in Programmiersprachen erworben. Sie sind in der Lage aktuelle Entwicklungen im Bereich der Programmiersprachen und des Compilerbaus zu beurteilen. Durch die Teilnahme an Programmierübungen mit Codereviews haben sie gelernt, qualitativ hochwertige Compiler zu entwickeln.</p>		
13. Inhalt:	<p>Lexer- und Parsergeneratoren, Semantische Attributierung, Fehlererkennung und -behandlung in Compilern, Typsysteme und Typprüfung, Die Java Virtual Machine, Zwischencodegenerierung, Sprachinterfaces</p>		
14. Literatur:	<p>A.W. Appel : Modern Compiler Implementation in Java 2nd Edition; Cambridge University Press (2002)</p> <p>A. V. Aho, M.S. Lahm, R. Sethi, J. D. Ullman: Compilers - Principles, Techniques, and Tools; Addison, Wesley (2007)</p>		
15. Lehrveranstaltungen und -formen:	581901 Vorlesung Entwurf und Implementierung eines Compilers		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden</p> <p>Selbststudium: 138 Stunden</p>		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 58191 Entwurf und Implementierung eines Compilers (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich 		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 56980 Fachpraktikum Cloud Architekturen und Management

2. Modulkürzel:	[pord.modulcode]	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Johannes Wettinger		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:	<p>Obligatory: Object-oriented programming (e.g. Java), XML Optional: scripting languages (e.g. Python, Ruby, Bash), JSON, YAML</p>		
12. Lernziele:	<p>The goal of this course is twofold: First, established reference architectures are discussed and used to model scalable architectures for Cloud applications. Second, based on these architectures, different Cloud management approaches are used to provision infrastructure resources, deploy application components, and manage them. The course is targeted at students of MSc Computer Science, MSc Softwaretechnik, and MSc Informatik.</p>		
13. Inhalt:	<p>Architectures for scalable, elastic, and reliable Cloud applications Cloud infrastructure management (OpenStack, Amazon Web Services) Configuration management (Chef, Puppet, CFEngine, etc.) Container virtualization (Docker, LXC, etc.) PaaS-centric management (Cloud Foundry, BOSH, etc.) Model-driven Cloud management: infrastructure-centric models (CloudFormation, Heat, etc.), application-centric models (Juju, TOSCA, etc.)</p>		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	569801 Fachpraktikum Cloud Architekturen und Management		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden</p>		
17. Prüfungsnummer/n und -name:	56981 Fachpraktikum Cloud Architekturen und Management (LBP), Studienbegleitend, 0 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29740 Fachpraktikum Eingebettete Systeme

2. Modulkürzel:	051711135	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	6.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	wiss. MA		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Modul "Embedded Systems Engineering"		
12. Lernziele:	Ability to apply the design methodology and commercial design tools for constructing and analyzing embedded hardware / software systems. Practical experience in software programming and debugging, digital circuit design and verification, usage of lab equipment such as logic analyzers. Experience in preparing structured technical documentation of specifications and designs.		
13. Inhalt:	This lab course focuses on analysis, design and implementation of embedded hardware/software systems and issues involved in the development of such systems. 1. Embedded software development 2. Usage of drivers for peripheral components 3. Cross-compilation 4. Remote debugging 5. Software performance profiling 6. Design of accelerator hardware digital circuits 7. Digital circuit simulation 8. FPGA implementation (synthesis) of digital circuits 9. Hardware / software interfacing 10. Integrated functional verification of hardware and software		
14. Literatur:	Lab handouts Documentation of development tools (provided in the lab)		
15. Lehrveranstaltungen und -formen:	297401 Übung Fachpraktikum Eingebettete Systeme		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit im Labor: 64 Stunden Vor- und Nachbereitung: 116 Stunden Summe: 180 Stunden		
17. Prüfungsnummer/n und -name:	29741 Fachpraktikum Eingebettete Systeme (LBP), schriftlich oder mündlich, Gewichtung: 1.0		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Eingebettete Systeme (Embedded Systems Engineering)

Modul: 29750 Fachpraktikum Rechnerarchitektur

2. Modulkürzel:	051700025	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Rafal Baranowski 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 41930 Rechnerorganisation • Modul 10140 Advanced Processor Architecture 		
12. Lernziele:	Students are able to design digital systems by using the complete state of the art design automation tool chain.		
13. Inhalt:	<p>In this lab course, the students design and implement a RISC processor and extend it with techniques common for high-performance processors. Hardware structures found in the state of the art processors will be applied and adapted.</p> <p>In order to achieve high frequency, proper design and verification techniques play an important role. The students learn how timing analysis, pipelining and retiming can be used to optimize the synthesis results. Because software has to be specifically tailored to such a processor architecture, the lab course also deals with scheduling techniques that allow to avoid pipeline stalls and hazards.</p>		
14. Literatur:	<ul style="list-style-type: none"> • D. A. Patterson, J. L. Hennessy: Computer Organization & Design . The Hardware / Software Interface (3rd Edition); San Francisco, Ca.: Morgan Kaufmann Publishers Inc., 2004 • J. L. Hennessy and D. A. Patterson: Computer Architecture - A Quantitative Approach (3rd Edition); San Francisco, Ca.: Morgan Kaufmann Publishers Inc., 2012 		
15. Lehrveranstaltungen und -formen:	297501 Fachpraktikum Rechnerarchitektur		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	29751 Fachpraktikum Rechnerarchitektur (LBP), schriftlich oder mündlich, Gewichtung: 1.0		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von: Rechnerarchitektur

Modul: 24900 Fachpraktikum Rechnergestützte Methoden der Produktentwicklung

2. Modulkürzel:	051400006	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes Semester
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dieter Roller		
9. Dozenten:	<ul style="list-style-type: none"> • Dieter Roller • Julian Eichhoff 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Grundkenntnisse wie sie in „CAD und Produktmodelle“ vermittelt werden.		
12. Lernziele:	<p>Die Studierenden sind nach erfolgreichem Besuch des Moduls in der Lage</p> <ul style="list-style-type: none"> • typische Problemstellungen, Methoden und Technologien im Anwendungsbereich des jeweils behandelten CAx-Feldes sowie dessen Eingliederung in die Produktentwicklungskette unter Verwendung von Fachvokabular zu beschreiben • die vorgestellten Methoden und Technologien gegenüberzustellen und ihren Einsatz im Bezug zu vorgegebenen Problemstellungen zu begründen • die Funktionen eines Technologievertreters aus dem CAx-Feld bei einer vorgegebenen Problemstellung effektiv anwenden können 		
13. Inhalt:	<p>Jedes Semester wechselnd werden zu einem konkreten CAx-Bereich (CAD, CAM, CAP, CAQ) folgende Inhalte behandelt:</p> <ul style="list-style-type: none"> • Eingliederung des Unternehmensbereichs hinsichtlich Produktentstehung/Produktlebenszyklus; Funktionen, Prozesse und Tätigkeiten des Unternehmensbereichs; Informationsflüsse (zugrundeliegende Daten, Objekte und Artefakte) und weitere Verbindung zu anderen Unternehmensbereichen • Überblick zu typischen Problemstellungen und darauf anwendbare Methoden im Aufgabenfeld des Unternehmensbereichs; Methodenvergleich • Überblick der marktdominierenden CAx-Technologien zur Unterstützung des Aufgabenfeldes; Technologievergleich • Anwendungsbezogene Detailvorstellung der Funktionen und Repräsentationsformen eines Vertreters dieser Technologien im Rahmen konkreter Beispielprobleme 		

14. Literatur:	<ul style="list-style-type: none">• D. Roller. CAD: Effiziente Anpassungs- und Variantenkonstruktion. Springer, Heidelberg, 1995.• S. Vajna, C. Weber, H. Bley, K. Zeman. CAx für Ingenieure - Eine praxisbezogene Einführung. Springer, Heidelberg, 2009.• G. Pahl, W. Beitz, J. Feldhusen, K.-H. Grote. Konstruktionslehre - Grundlagen erfolgreicher Produktentwicklung Methoden und Anwendung. Springer, Heidelberg, 2007.
15. Lehrveranstaltungen und -formen:	249001 Übung Fachpraktikum Rechnergestützte Methoden der Produktentwicklung
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	24901 Fachpraktikum Rechnergestützte Methoden der Produktentwicklung (LBP), schriftlich oder mündlich, Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

Modul: 45750 Fachpraktikum Verteilte Systeme

2. Modulkürzel:	051200111	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Univ.-Prof. Kurt Rothermel	
9. Dozenten:		Frank Dürr	
10. Zuordnung zum Curriculum in diesem Studiengang:		<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>	
11. Empfohlene Voraussetzungen:		Verteilte Systeme, Rechnernetze II	
12. Lernziele:		<p>Die Teilnehmer besitzen die Fähigkeit, verteilte Anwendungen und Dienste zu entwerfen und zu implementieren. Sie besitzen praktische Kenntnisse in der Netzprogrammierung und der Programmierung von Client/Server-Anwendungen.</p> <p>Sie verfügen über praktische Kenntnisse über Technologien und Werkzeugen zur Implementierung und zum Testen verteilter Systeme.</p>	
13. Inhalt:		<ul style="list-style-type: none"> - Socket-Programmierung - Höherwertige Kommunikationsprotokolle und Webservice- Schnittstellen (HTTP & XML/JSON, RPC, SOAP, REST) - Client/Server-Systeme - Peer-to-Peer- und Ad-hoc-Kommunikation - Entwicklungsumgebungen - Test verteilter Systeme 	
14. Literatur:		- A.S. Tanenbaum: Computer Networks, 4th Edition, 2003	
15. Lehrveranstaltungen und -formen:		457501 Fachpraktikum Verteilte Systeme	
16. Abschätzung Arbeitsaufwand:		Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden	
17. Prüfungsnummer/n und -name:		45751 Fachpraktikum Verteilte Systeme (PL), Sonstiges, Gewichtung: 1.0	
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:		Verteilte Systeme	

Modul: 58440 Fachpraktikum: Algorithmik

2. Modulkürzel:	-	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Univ.-Prof. Stefan Funke	
9. Dozenten:			
10. Zuordnung zum Curriculum in diesem Studiengang:		B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →	
11. Empfohlene Voraussetzungen:			
12. Lernziele:			
13. Inhalt:			
14. Literatur:			
15. Lehrveranstaltungen und -formen:	584401 Fachpraktikum Algorithmik		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	58441 Fachpraktikum: Algorithmik (LBP), Sonstiges, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29440 Geometric Modeling and Computer Animation

2. Modulkürzel:	051900010	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. Daniel Weiskopf		
9. Dozenten:	<ul style="list-style-type: none"> • Daniel Weiskopf • Thomas Ertl • Guido Reina 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>Basic computer graphics, for example:</p> <ul style="list-style-type: none"> • 10060 Computergraphik 		
12. Lernziele:	<p>Students gain an understanding of the fundamental concepts and techniques of geometric modeling and computer animation. This includes theoretical and mathematical foundations, important algorithms, and implementation aspects as well as practical experience with modeling and animation tools such as Maya.</p>		
13. Inhalt:	<p>This course covers foundations and methods for the modeling of scenes and for computer animation. This includes the representation of curves and surfaces, which are used by modeling and animation software for modeling of objects, description of the dynamics of parameters, or keyframe animation. Physically based animation describes motion via kinematic and dynamics laws of mechanics. Applications thereof include particle systems all the way to character animation and deformation.</p> <p>In particular, the following topics are covered:</p> <ul style="list-style-type: none"> • Description and modeling of curves: differential geometry of curves, polynomial curves in general, interpolation, Bezier curves, B-splines, rational curves, NURBS • Description and modeling of surfaces: differential geometry of surfaces, tensor product surfaces, Bezier patches, NURBS, ruled surfaces, Coons paths • Subdivision schemes: basic concept, convergence and limit process, subdivision curves, subdivision surfaces • Overview of animation techniques • Keyframe animation, inverse kinematics 		

- Physically based animation of points and rigid bodies: kinematics and dynamics
- Particle systems: Reeves, flocking and boids, agent-based simulation
- Cloth animation: continuum mechanics, mass-spring model, numerical solvers for ordinary differential equations, explicit and implicit integrators
- Collision: efficient collision detection, bounding volume hierachies, hierarchical space partitioning, collision handling, sliding and resting contact
- Fluid simulation: wave equation, Navier Stokes, level sets, particle level sets
- Basics of film production: camera, lighting, production process, storyboard

14. Literatur:

- D. Eberly, 3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics. Morgan Kaufmann, 2000
- G. Farin: Curves and Surfaces for CAGD: A Practical Guide. Morgan Kaufmann, 2002
- R. Parent: Computer Animation: Algorithms and Techniques. Morgan Kaufmann, 2002
- W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterling: Numerical Recipies - The Art of Scientific Computing. Cambridge University Press, 1986

15. Lehrveranstaltungen und -formen:

294401 Vorlesung mit Übungen Geometrische Modellierung und Animation

16. Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden, Selbststudium: 138 Stunden

17. Prüfungsnummer/n und -name:

- 29441 Geometric Modeling and Computer Animation (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0,
- V Vorleistung (USL-V), schriftlich, eventuell mündlich, Erfolgreiche Teilnahme an Übungen / exercises passed

18. Grundlage für ... :

19. Medienform:

Video projector, blackboard, exercises using PCs

20. Angeboten von:

Institut für Visualisierung und Interaktive Systeme

Modul: 29450 Graphentheorie

2. Modulkürzel:	050420105	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	<ul style="list-style-type: none"> • Ulrich Hertrampf • Volker Diekert • Manfred Kufleitner 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Grundvorlesungen in theoretischer Informatik		
12. Lernziele:	Die Studierenden beherrschen typische Denk- und Herangehensweisen aus der Graphentheorie. Die Beziehung zwischen diversen Graphparametern werden verstanden, ebenso wie ihre algorithmische Relevanz. Die Eigenschaften der wichtigsten Graphklassen erschließen sich den Studierenden.		
13. Inhalt:	<p>Die Vorlesung behandelt algorithmische Problem und strukturelle Zusammenhänge bei Graphen. Im Einzelnen werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none"> • Eulergraphen • Cographen • Bipartite Graphen • Planare Graphen, Eulerformel, Satz von Kuratowski • Graphparameter • Perfekte Graphen • Graphenfärbungen und der Satz von Ramsey • Extremale Graphentheorie 		
14. Literatur:	<ul style="list-style-type: none"> • Reinhard Diestel: Graphentheorie. Springer, 2010. • Martin Aigner, Günter M. Ziegler: Das BUCH der Beweise. Springer, 2009. • Jacobus H. van Lint, Richard M. Wilson: A Course in Combinatorics. Cambridge University Press, 2nd edition, 2001. 		
15. Lehrveranstaltungen und -formen:	294501 Vorlesung mit Übungen Graphentheorie		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Gesamt: 180 h		

Modul: 46430 Grundlagen der Betriebswirtschaftslehre

2. Modulkürzel:	100110001	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	unregelmäßig
4. SWS:	3.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Wolfgang Burr		
9. Dozenten:	<ul style="list-style-type: none"> • Wolfgang Burr • Torsten Frohwein • Xenia Schmidt 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<ul style="list-style-type: none"> • Die Studierenden können die zentrale betriebswirtschaftliche Definitionen wiedergeben und lernen auf deren Basis zu argumentieren • Die Studierenden können die verschiedene Teilbereiche der Betriebswirtschaft benennen und in das Gesamtkonzept der Betriebswirtschaft einordnen sowie dortige Problemstellungen angeben und eingesetzte Instrumente anwenden • Die Studierenden sind in der Lage ausgewählte betriebswirtschaftlichen Theorien zu erklären und auf bestimmte Problemstellungen anzuwenden 		
13. Inhalt:	<p>Dieses einführende Modul bringt zunächst den Studierenden den Gegenstand der Betriebswirtschaftslehre näher und ermöglicht ein Kennenlernen erster betriebswirtschaftlicher Begriffe sowie eine Einordnung der Betriebswirtschaftslehre in den Rahmen der Wirtschaftswissenschaften.</p> <p>Weiterhin werden die entscheidungstheoretischen Grundlagen und Modelle diskutiert. Anhand praxisorientierter Aufgaben wird die Entscheidungsproblematik begrifflich gemacht. Ferner werden die Einheiten der betrieblichen Leistungserstellung und die Instrumente zur Unterstützung dieser erläutert.</p> <p>Schließlich lernen die Studierenden die Aufgaben und Probleme der Unternehmensführung kennen. Neben der Einführung in die Theorien, Methoden und Konzepte der Unternehmensführung, bekommen die Studierenden Einblick in weitere Bereiche wie z. B. Innovationsmanagement.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Folien zu Vorlesungen und Übungen • Übungsaufgaben im ILIAS <p>Die Basisliteratur umfasst die folgenden Werke:</p>		

	<ul style="list-style-type: none"> • Burr, W.: Innovationen in Organisationen, aktuelle Auflage, Kohlhammer Verlag, Stuttgart. • Burr, W., Musil, A., Stephan, M., Werkmeister, C.: Unternehmensführung, aktuelle Auflage, Verlag Vahlen, München. • Thommen, J.-P., Achleitner, A.-K.: Allgemeine Betriebswirtschaftslehre, aktuelle Auflage, Springer, Gabler Verlag, Wiesbaden.
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 464301 Vorlesung Grundlagen der Betriebswirtschaftslehre • 464302 Übung Grundlagen der Betriebswirtschaftslehre
16. Abschätzung Arbeitsaufwand:	<p>Vorlesung</p> <ul style="list-style-type: none"> - Präsenzzeit: 28 h - Selbststudium: 32 h <p>Übung</p> <ul style="list-style-type: none"> - Präsenzzeit: 14 h - Selbststudium: 16 h <p>Gesamt: 90 h</p>
17. Prüfungsnummer/n und -name:	46431 Grundlagen der Betriebswirtschaftslehre (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	ABWL, Forschungs-, Entwicklungs- und Innovationsmanagement

Modul: 41980 Grundlagen der VWL

2. Modulkürzel:	100402007	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	3.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Bernd Woeckener		
9. Dozenten:	Bernd Woeckener		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Studierenden sind nach Abschluss des Moduls in der Lage, <ul style="list-style-type: none"> • auf der Basis der zentralen ökonomischen Begrifflichkeiten und Konzepte zu argumentieren, • das Funktionieren und die Funktionsbedingungen von Märkten richtig einzuschätzen, • auf der Basis der Kenntnis der wichtigsten makroökonomischen Größen und ihrer Zusammenhänge gesamtwirtschaftliche Argumentationen und Politikansätze kompetent einzuschätzen. 		
13. Inhalt:	Dieses einführende Modul behandelt die grundlegenden Begriffe, Konzepte und Methoden der einzel- und marktwirtschaftlichen (mikroökonomischen) sowie der gesamtwirtschaftlichen (makroökonomischen) Theorie. Aufbauend auf den grundlegenden Konzepten der Knappheit, der Kosten und der Arbeitsteilung steht im mikroökonomischen Teil das Funktionieren von Märkten als Orten des Aufeinandertreffens von Angebot und Nachfrage im Mittelpunkt. Der makroökonomische Teil erläutert die zentralen gesamtwirtschaftlichen Größen (Aggregate) einer offenen Volkswirtschaft und analysiert die Zusammenhänge zwischen diesen Größen.		
14. Literatur:	<ul style="list-style-type: none"> • B. Woeckener: Volkswirtschaftslehre, Springer, neueste Auflage • P. Samuelson: Economics, McGraw-Hill/ Irwin, neueste Auflage 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 419801 Vorlesung Einführung in die VWL • 419802 Übung Einführung in die VWL 		
16. Abschätzung Arbeitsaufwand:	Vorlesung: Präsenzzeit: 28 h Selbststudiumszeit / Nacharbeitszeit: 84 h Übung: Präsenzzeit: 14 h Selbststudiumszeit / Nacharbeitszeit: 42 h Gesamt: 168 h		
17. Prüfungsnummer/n und -name:	41981 Grundlagen der VWL (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Mikroökonomik und räumliche Ökonomik

Modul: 29610 Hardware Based Fault Tolerance

2. Modulkürzel:	051710023	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Michael Kochte 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10140 Grundlagen der Rechnerarchitektur / Advanced Processor Architecture • Modul 10310 Rechnerorganisation 		
12. Lernziele:	<ul style="list-style-type: none"> • Knowledge of methods for reliability assessment of circuits and systems • Knowledge of the main techniques for implementing fault tolerance • Knowledge how to design fault tolerant circuits and systems 		
13. Inhalt:	<p>Micro- and Nano-electronic systems can exhibit failures both right after production and during their operation. Systems for which safety and security is of concern have to be designed in a way that the desired function can be delivered even if some components fail or produce erroneous outputs. This lecture presents the most important design techniques that allow to tolerate hardware faults up to a certain degree. The topics of the lecture are as follows:</p> <ul style="list-style-type: none"> • Terminology • Measures of fault tolerance • Techniques for structural and time redundancy • Error detection and diagnosis • Fault masking, repair, reconfiguration • Fault-tolerant distributed systems 		
14. Literatur:	<p>Apart from lecture slides, the following books can be used to deepen on the topics of the lecture:</p> <ul style="list-style-type: none"> • I. Koren and C. M. Krishna: Fault-Tolerant Systems, Morgan-Kaufman (2007) • P. K. Lala: Self-Checking and Fault-Tolerant Digital Design, Morgan-Kaufmann Publishers (2001) • D.K. Pradhan: Fault-Tolerant Computer Design, Prentice Hall (1996) 		

	<ul style="list-style-type: none">• R.N. Rao and E. Fujiwara: Error Control Coding for Computer Systems, Prentice Hall (1989)• M.L. Bushnell and V.D. Agrawal: Essentials of Electronic Testing, Klumer Academic Publishers (2000)
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 296101 Vorlesung Hardware Based Fault Tolerance• 296102 Übung Hardware Based Fault Tolerance
16. Abschätzung Arbeitsaufwand:	Presence Time: 42 Stunden Self Study: 138 Stunden Sum: 180 Stunden
17. Prüfungsnummer/n und -name:	29611 Hardware Based Fault Tolerance (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0, Writen exam 90 min or Oral exam 30 min
18. Grundlage für ... :	
19. Medienform:	Laptop presentation
20. Angeboten von:	Institut für Technische Informatik

Modul: 14380 Hardware Verification and Quality Assessment

2. Modulkürzel:	051700020	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Michael Kochte 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 4. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10310 Rechnerorganisation oder • Modul 10140 Grundlagen der Rechnerarchitektur 		
12. Lernziele:	Basic knowledge of methodologies and algorithms of functional and formal verification, diagnosis, test and design for testability of integrated circuits		
13. Inhalt:	<p>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.</p> <p>The course comprises:</p> <ul style="list-style-type: none"> • Validation: Simulation and emulation in different design levels. • Formal verification: Equivalence checking and model checking. • Test: Fault simulation and test generation. • Debug and diagnosis. 		
14. Literatur:	<ul style="list-style-type: none"> • 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 • S. Hassoun, T. Sasao: Logic Synthesis and Verification, 2002 		

	<ul style="list-style-type: none">• S. Minato: Binary Decision Diagrams and Applications for VLSI CAD, 1996• T. Kropf: Introduction to Formal Hardware Verification, 1999
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 143801 Vorlesung Hardware Verification and Quality Assessment• 143802 Übung Hardware Verification and Quality Assessment
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	14381 Hardware Verification and Quality Assessment (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Technische Informatik

Modul: 42920 Hardware-Software-Codesign

2. Modulkürzel:	051711110	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Bachelor-Veranstaltung "Grundlagen der Eingebetteten Systeme" oder gleichwertige Kenntnisse		
12. Lernziele:	Ability to conceptualize systems so that an application-specific, optimized trade-off between hardware and software implementation of system functionality is achieved.		
13. Inhalt:	<p>This module deals with the joint design and optimization of hardware and software for pre-defined applications, covering the following topics:</p> <ol style="list-style-type: none"> 1. Models for system specification 2. Modelling and simulation with the SystemC library 3. Synthesis of system architectures 4. Resource allocation and operation binding 5. Partitioning of functionality among hardware and software 6. Scheduling and schedulability for parallel multi-core architectures 7. Methods for system optimization 8. Application specific instruction set processors (ASIPs) 9. Network-on-Chip (NoC) interconnect architectures 		
14. Literatur:	J. Teich, Digitale Hardware/Software-Systeme, 2. Auflage, 2007		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 429201 Vorlesung Hardware-Software-Codesign • 429202 Übung Hardware-Software-Codesign 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	42921 Hardware-Software-Codesign (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Eingebettete Systeme (Embedded Systems Engineering)		

Modul: 40620 Hauptseminar INF 1

2. Modulkürzel:	050420195	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:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->Hauptseminar →		
11. Empfohlene Voraussetzungen:	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. 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:	406201 Hauptseminar (Master INF)		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden Gesamt: 90 Stunden		
17. Prüfungsnummer/n und -name:	40621 Hauptseminar INF 1 (BSL), schriftlich oder mündlich, Gewichtung: 1.0		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 43060 Hauptseminar INF 2

2. Modulkürzel:	050420195	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:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->Hauptseminar →		
11. Empfohlene Voraussetzungen:	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. 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:	430601 Hauptseminar		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden Gesamt: 90 Stunden		
17. Prüfungsnummer/n und -name:	43061 Hauptseminar INF 2 (BSL), schriftlich oder mündlich, Gewichtung: 1.0		

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 42420 High Performance Computing

2. Modulkürzel:	051240040	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Martin Bernreuther • Dirk Pflüger • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 6. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • 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:	<ul style="list-style-type: none"> • Fähigkeit, parallele Algorithmen auf unterschiedlichen parallelen Plattformen mit Hilfe geeigneter algorithmischer Modelle zu bewerten. • Kenntnis verschiedener Programmiermodelle für Parallelrechner mit verteiltem und gemeinsamem Speicher. • Fähigkeit, auch fortgeschrittene Implementierungsaufgaben aus dem Bereich des Höchstleistungsrechnens auf Basis ausgewählter Programmiermodelle zu bewältigen. 		
13. Inhalt:	<p>Die Vorlesung beschäftigt sich mit den Grundlagen paralleler Programmierung und paralleler Algorithmen speziell im Hinblick auf die Anwendungsbereiche Wissenschaftliches Rechnen und High Performance Computing.</p> <p>Verwandte Fragestellungen aus dem Bereich der Theorie (parallele Modelle und parallele Komplexität, etc.) sowie aus der Rechnertechnik (parallele Architekturen) werden begleitend diskutiert.</p>		

Nach einer allgemeinen Einführung (Klassifizierung von Parallelrechnern, Ebenen von Parallelität, Performance und Architekturen, etc.), werden die Grundlagen paralleler Programme eingeführt (Notation/Syntax, Synchronisation und Kommunikation, Design paralleler Programme, etc.). Sowohl die Programmierung auf Systemen mit gemeinsamem Speicher als auch auf Systemen mit verteiltem Speicher werden besprochen. Dabei wird jeweils mindestens ein geeignetes Programmiermodell (z.B. OpenMP, MPI, CUDA) vertieft behandelt. Aus dem Bereich des High Performance Computing werden begleitend klassische Algorithmen und Implementierungstechniken als Beispiele behandelt, z.B. parallele Algorithmen aus der linearen Algebra (Matrixmultiplikation, etc. oder einfache Verfahren zur Lösung partieller Differentialgleichungen). Zusätzlich können Themen wie Lastverteilung und Lastbalancierung (Grundlagen, Algorithmen zur Partitionierung und Lastbalancierung, etc.) vorgestellt werden.

14. Literatur:	<ul style="list-style-type: none"> • T. Rauber, G. Rüniger: „Parallele Programmierung“, 2. Aufl., Springer 2007; (in English: T. Rauber, G. Rüniger: „Parallel Programming: for Multicore and Cluster Systems“, Springer 2010) • K.A. Berman, J.L. Paul: "Sequential and Parallel Algorithms", PWS Publishing Company, 1997 • B. Chapman, G. Jost, R. van der Pas: "Using OpenMP - Portable Shared Memory Parallel Programming", MIT Press, 2008 • W. Gropp, E. Lusk, und R. Thakur: "Using MPI-2: Advanced Features of the Message-Passing Interface", das Buch ist auch in deutscher Übersetzung erhältlich. • D. Kirk, W.-M. Hwu Programming Massively Parallel Processors
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 424201 Vorlesung High Performance Computing • 424202 Übung High Performance Computing
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Selbststudiumszeit: 138 Stunden</p>
17. Prüfungsnummer/n und -name:	<p>42421 High Performance Computing (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0</p>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Simulation großer Systeme

Modul: 51720 IT-Strategy

2. Modulkürzel:	-	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. Frank Leymann		
9. Dozenten:	Sven Lorenz		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>Die Vorlesung fokussiert auf Management Strategien. Es wird erläutert, wie solche Strategien entwickelt und evaluiert werden.</p> <p>Teilnehmer der Vorlesung verstehen die Bestandteile einer IT Strategie. Sie können eine IT Strategie ableiten und entwickeln, basierend auf dem aktuellen Status eines Unternehmens. Insbesondere wird verstanden, was unter den Begriffen und Konzepten IT Organisation, Sourcing Management, Architektur Management, Qualitäts- und Risk-Management und schliesslich IT Landschaften zu verstehen ist und wie man damit umgeht.</p>		
13. Inhalt:	<p>Über die Einstiegsfragestellung „Was ist ‚Strategie‘?“ wird erläutert, was eine Unternehmensstrategie und eine IT-Strategie ist, wobei sowohl die klassischen Ansätze als auch neue Sichtweisen vorgestellt werden. Im Schwerpunkt „Strategieentwicklung“ wird auf die Ableitung der IT-Strategie aus der Unternehmensstrategie eingegangen. Ein kanonisches Vorgehensmodell wird eingeführt und anhand von Unternehmensbeispielen illustriert. Der Schwerpunkt „IT-Strategie als Prozess“ beginnt mit der Einbettung der IT-Strategieaufgaben in die bekannten IT Prozessmodelle wie ITIL und CobiT. Im Rahmen eines verallgemeinerten IT-Prozessmodells werden die einzelnen IT-Strategieprozesse (IT-Organisationsentwicklung, IT-Sourcing-Strategie, IT-Architektur-Management, IT-Bebauungsplanung, IT-Qualitätsmanagement und IT-Risikomanagement) in der Folge detailliert erläutert. Dabei werden klassische und State-of-the-art Methoden und Werkzeuge zur Unterstützung der IT-Strategieprozesse vorgestellt. Exkurse in das IT-Portfoliomanagement und in IT-Kennzahlensysteme runden die Vorlesungsinhalte ab.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Helmut Krcmar, „Informationsmanagement“, Springer, 2010 		

- Jürgen Hofmann, Werner Schmitt, „Masterkurs IT-Management“, VIEWEG+TEUBNER, 2010W.
- Brenner, A. Resch, V. Schulz, „Die Zukunft der IT in Unternehmen“, FAZ Buch, 2010
- Martin Kütz, „Kennzahlen in der IT“, dpunkt-Verlag, 2007

15. Lehrveranstaltungen und -formen:	517201 Vorlesung mit Übungen IT-Strategie
16. Abschätzung Arbeitsaufwand:	Lecture & exercises: 42 hours Self-study: 138 hours
17. Prüfungsnummer/n und -name:	51721 IT-Strategy (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Architektur von Anwendungssystemen

Modul: 48500 Image Synthesis

2. Modulkürzel:	051903654	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Martin Fuchs		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Martin Fuchs 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Modul 10060 Computergraphik		
12. Lernziele:	<p>The students know the theoretical foundations of image synthesis and have practical expertise in programming of rendering systems. They know several approaches and algorithms for three-dimensional computer graphics, both for real-time and physically accurate rendering.</p>		
13. Inhalt:	<p>The class covers physically based rendering techniques such as ray/path tracing and radiosity, computer graphics models for light transport and light/scene interaction, as well as numerical methods such as Monte Carlo integration and finite element methods which approximate solutions to the rendering equation. In addition, techniques which specifically employ modern graphics processing hardware are covered which approximate physically correct solutions in interactive application scenarios by means of rasterization and image-space rendering.</p> <p>Specifically, the class covers:</p> <ul style="list-style-type: none"> • graphics hardware and rasterization APIs by example of OpenGL • textures and procedural models • shading and shadow computations in rasterization pipelines • scene graphs, culling and level-of-detail approaches • physically based rendering and photo-realistic image synthesis • local shading and material models, especially the BRDF • the rendering equation • ray tracing and Monte-Carlo approaches • global illumination simulation (especially by means of radiosity, distribution ray tracing and path tracing) 		
14. Literatur:	<ul style="list-style-type: none"> • Andrew S. Glassner, Principles of Digital Image Synthesis, 1995J. • Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990M. 		

	<ul style="list-style-type: none">• Pharr, G. Humphreys, Physically Based Rendering, 2004
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 485001 Lecture Image Synthesis• 485002 Exercise Image Synthesis
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 48501 Image Synthesis (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0• V Vorleistung (USL-V), schriftlich oder mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 51540 Implementierung Finiter Elemente

2. Modulkürzel:	080803884	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Claus-Justus Heine		
9. Dozenten:	Claus-Justus Heine		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:	empfohlen: „Einführung in die Numerik partieller Differentialgleichungen“ oder „Partielle Differentialgleichungen (Modellierung, Analysis, Simulation)“		
12. Lernziele:	<ul style="list-style-type: none"> • Umgang mit gebräuchlichen Finite-Elemente Toolboxen Praktische Umsetzung von Finite-Elemente • Methoden am Computer Validierung der Implementierung anhand der theoretischen • Vorhersagen Darstellung und Visualisierung von Simulationsergebnissen 		
13. Inhalt:	Die Studierenden erwerben die Fähigkeit, die Diskretisierung partieller Differentialgleichungen mit adaptiven Finite-Elemente Verfahren praktisch am Computer umzusetzen. Die Umsetzung am Computer erfolgt im Rahmen einer gebräuchlichen Finite Elemente Toolbox (z.B. DUNE). Teil der praktischen Umsetzung ist die experimentelle Validierung der numerischen Verfahren und die Visualisierung der Simulationsergebnisse. Die numerischen Verfahren bauen auf den theoretischen Kenntnissen auf, die zum Beispiel in einer der beiden empfohlenen vorangehenden Vorlesungen erworben werden können.		
14. Literatur:	Schmidt, A. & Siebert, K. G.: Design of adaptive finite element software Springer, 2005, 42, XII. Braess, D.: Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie, Springer Spektrum, 2013, XVI. Brenner, S. C.; Scott, L. R.: The mathematical theory of finite element methods, Springer, 2010, XVII. Weitere Titel nach Bekanntgabe in der Vorlesung		
15. Lehrveranstaltungen und -formen:	515401 Vorlesung und Übung Implementierung Finiter Elemente		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42h Selbststudium/Nacharbeitszeit: 118h Projektvorstellung mit Vorbereitung: 20h Gesamt: 180h		

17. Prüfungsnummer/n und -name: 51541 Implementierung Finiter Elemente (BSL), schriftliche Prüfung,
Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 55610 Information Integration

2. Modulkürzel:	051210166	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Melanie Herschel		
9. Dozenten:	Melanie Herschel		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Lecture "Modellierung" or comparable course		
12. Lernziele:	Integrating heterogeneous, autonomous and structured data is essential in an interconnected world. This is the basis for information exchange and comprehensive search. The goal of this course is to provide an overview of challenges in information integration and to enable the students to assess available approaches and technologies.		
13. Inhalt:	Based on application scenarios from various organizations, we will discuss aspects of distribution, autonomy and heterogeneity. This helps us to organize the problem space and to compare possible architectures of integrated information systems. Heterogeneity is addressed by schema mappings between and data mappings. We will discuss how to establish such mappings and how to apply them in data transformation. As query processing in federated databases is based on these mappings as well, we will also learn the basics on these systems. Another focus of this course is on the pre-processing and integration of data. Starting with a discussion on information quality, we will look at the spectrum of erroneous data and approaches to data cleansing. State-of-the-art software for information integration will be presented, in particular as part of the exercises.		
14. Literatur:	<p>Additional literature will be announced at the beginning of the lecture</p> <ul style="list-style-type: none"> • Ulf Leser, Felix Naumann: Informationsintegration: Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen, dpunkt Verlag, 2006, ISBN 3898644006. 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 556101 Vorlesung Information Integration • 556102 Übung Information Integration 		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden</p> <p>Selbststudium: 138 Stunden</p>		

Gesamt: 180 Stunden

17. Prüfungsnummer/n und -name: 55611 Information Integration (PL), schriftlich oder mündlich, 60 Min.,
Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Datenbanken und Informationssysteme

Modul: 55630 Information Visualization and Visual Analytics

2. Modulkürzel:	051900099	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Daniel Weiskopf		
9. Dozenten:	<ul style="list-style-type: none"> • Steffen Koch • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Basic Human Computer Interaction		
12. Lernziele:	Student gains expertise about fundamental concepts and techniques of information visualization and visual analytics. This includes algorithms and mathematical background, data structures and implementation aspects as well as practical experience with widely available visualization tools.		
13. Inhalt:	<p>Topics covered in this course:</p> <ul style="list-style-type: none"> - Perception and Cognition - Graphs and Networks - Hierarchies and Trees - Multi-dimensional and high-dimensional data visualization - Time series visualization - Visual Analytics - Software Visualization - Geospatial visualization 		
14. Literatur:	<ul style="list-style-type: none"> • Colin Ware. Visual Thinking for Design • Colin Ware. Information Visualization. Perception for Design • Edward Tufte. The Visual Display of Quantitative Information • Robert Spence. Design for Interaction • Jim Thomas. Illuminating the Path 		
15. Lehrveranstaltungen und -formen:	556301 Vorlesung und Übung Informationsvisualisierung		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

Gesamt: 180 Stunden

17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 55631 Information Visualization and Visual Analytics (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0• V Vorleistung (USL-V), schriftlich, eventuell mündlich, Erfolgreiche Übungsteilnahmen / excercises passed
18. Grundlage für ... :	
19. Medienform:	Video projector, blackboard, exercises using PCs
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 60120 Interaktive Systeme

2. Modulkürzel:	-	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes Semester
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Albrecht Schmidt		
9. Dozenten:	<ul style="list-style-type: none"> • Albrecht Schmidt • Niels Henze 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009 → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012 → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012 → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012 → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012 → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Vorlesung Mensch-Computer Interaktion		
12. Lernziele:	Die Studierenden lernen, wie interaktive Systeme entwickelt werden. Sie verstehen den Entwicklungsprozess und können interaktive Systeme für spezifische Plattform entwickeln.		
13. Inhalt:			
14. Literatur:			
15. Lehrveranstaltungen und -formen:	601201 Fachpraktikum Interaktive Systeme		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 56 Stunden Selbststudium: 124 Stunden		
17. Prüfungsnummer/n und -name:	60121 Interaktive Systeme (LBP), Sonstiges, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29420 Konkrete Mathematik

2. Modulkürzel:	050420120	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	<ul style="list-style-type: none"> • Volker Diekert • Ulrich Hertrampf • Manfred Kufleitner 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->TMG-INF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Studierenden kennen die grundlegenden Ergebnisse und Methoden der konkreten Mathematik.		
13. Inhalt:	Behandelt werden moderne Teilgebiete der modularen Arithmetik, diskreten Mathematik, erzeugende Funktionen und Kombinatorik.		
14. Literatur:	<p>Volker Diekert, Manfred Kufleitner, Gerhard Rosenberger: Elemente der Diskreten Mathematik, Walter de Gruyter, 2013.</p> <p>Volker Diekert, Manfred Kufleitner, Gerhard Rosenberger: Diskrete algebraische Methoden, Walter de Gruyter, 2013.</p> <p>Ronald L. Graham, Donald E. Knuth, Oren Patshnik: Concrete Mathematics: A Foundation for Computer Science, Addison-Wesley, 1994.</p>		
15. Lehrveranstaltungen und -formen:	294201 Vorlesung mit Übungen Konkrete Mathematik		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden.</p> <p>Nachbearbeitungszeit: 138 Stunden.</p> <p>Gesamtzeit: 180 Stunden.</p>		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29421 Konkrete Mathematik (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftliche Prüfung, 120 Min. 		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von:

Theoretische Informatik

Modul: 29460 Kryptographische Verfahren

2. Modulkürzel:	050420110	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	<ul style="list-style-type: none"> • Ulrich Hertrampf • Volker Diekert • Stefan Funke 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Theorie-Vorlesungen des Bachelor-Studiums		
12. Lernziele:	Die Studierenden kennen die wichtigsten Sätze der Kryptographie. Sie können klassische und moderne Verschlüsselungsverfahren anwenden und die Sicherheit dieser Verfahren beurteilen und einstufen.		
13. Inhalt:	Moderne Verfahren der einstigen "Geheimwissenschaft" Kryptographie werden eingeführt. Die Veranstaltung stellt Methoden zur Erzeugung elektronischer Unterschriften und zur Identifikation von Benutzern vor, die als notwendige Voraussetzungen für elektronische Wahlen oder anonymes elektronisches Bargeld gelten. Es werden neben klassischen, symmetrischen Verschlüsselungsverfahren aktuelle asymmetrische Verfahren behandelt. Eine wichtige Rolle spielen Protokolle, die aufbauend auf kryptographischen Verfahren die erwähnten Aufgaben lösen.		
14. Literatur:	<ul style="list-style-type: none"> • Bruce Schneier, Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C, 1996 • Douglas Robert Stinson, Cryptography: Theory and Practice, 1995 • Friedrich Ludwig Bauer, Entzifferte Geheimnisse: Methoden und Maximen der Kryptologie, 1995 • Johannes Buchmann, Einführung in die Kryptographie, 1999 		
15. Lehrveranstaltungen und -formen:	294601 Vorlesung mit Übungen Kryptographische Verfahren		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29461 Kryptographische Verfahren (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich oder mündlich 		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Theoretische Informatik

Modul: 29480 Loose Coupling and Message Based Applications

2. Modulkürzel:	052010003	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Frank Leymann • Dimka Karastoyanova 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Understand the problem of application integration and the fundamental concept of loose coupling. The pros and cons of messaging are clear, and the architecture of Message Oriented Middleware is understood. Key patterns of using messaging to solve (enterprise) application integration problems are understood.		
13. Inhalt:	<p>Messaging is a cornerstone of the integration of heterogeneous applications inside and among enterprises. Applications that need to share data synchronously or asynchronously with each other can be made to interoperate by means of the feature-rich Message-Oriented Middleware (MOM) that has grown ubiquitous in enterprises. During this course we treat the approaches and challenges of application integration through messaging. At first, we will address concepts such as (a-)synchronous messaging and the different messaging styles, e.g. point-to-point and publish-subscribe, that are the foundation of message-based application integration. Later in the course we will take an in-depth look at the mechanics and architecture of MOM, in particular of the Java Messaging Service (JMS), which will also be used in examples and exercises. Throughout the course we will discuss and apply extensively Enterprise Application Integration (EAI) patterns. Especially, endpoint patterns, routing patterns, transformation patterns, messaging patterns, channel patterns, and management patterns will be presented; the composability of these patterns will be explained.</p>		
14. Literatur:	<p>G. Hohpe and B. Woolf: "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions." Addison-Wesley Professional, ISBN-13: 978-0321200686. October 2003.</p> <p>M. Hapner et al: "Java Messaging Service API Tutorial & Reference". Addison-Wesley 2001.</p>		

15. Lehrveranstaltungen und -formen:	294801 Vorlesung mit Übungen Lose Kopplung & Message-basierte Integration
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 29481 Loose Coupling and Message Based Applications (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0• V Vorleistung (USL-V), Sonstiges
18. Grundlage für ... :	
19. Medienform:	Lecture and accompanying exercises
20. Angeboten von:	Architektur von Anwendungssystemen

Modul: 29470 Machine Learning

2. Modulkürzel:	051200112	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Marc Toussaint		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Solid knowledge in Linear Algebra, probability theory and optimization. Fluency in at least one programming language.		
12. Lernziele:	Students will acquire an in depth understanding of Machine Learning methods. The concepts and formalisms of Machine Learning are understood as generic approach to a variety of disciplines, including image processing, robotics, computational linguistics and software engineering. This course will enable students to formalize problems from such disciplines in terms of probabilistic models and the derive respective learning and inference algorithms.		
13. Inhalt:	<p>Exploiting large-scale data is a central challenge of our time. Machine Learning is the core discipline to address this challenge, aiming to extract useful models and structure from data. Studying Machine Learning is motivated in multiple ways: 1) as the basis of commercial data mining (Google, Amazon, Picasa, etc), 2) a core methodological tool for data analysis in all sciences (vision, linguistics, software engineering, but also biology, physics, neuroscience, etc) and finally, 3) as a core foundation of autonomous intelligent systems (which is my personal motivation for research in Machine Learning).</p> <p>This lecture introduces to modern methods in Machine Learning, including discriminative as well as probabilistic generative models. A preliminary outline of topics is:</p> <ul style="list-style-type: none"> • motivation and history • probabilistic modeling and inference • regression and classification methods (kernel methods, Gaussian Processes, Bayesian kernel logistic regression, relations) • discriminative learning (logistic regression, Conditional Random Fields) • feature selection • boosting and ensemble learning 		

- representation learning and embedding (kernel PCA and derivatives, deep learning)
- graphical models
- inference in graphical models (MCMC, message passing, variational)
- learning in graphical models
- structure learning and model selection
- relational learning

Please also refer to the course web page: <http://ipvs.informatik.uni-stuttgart.de/mlr/marc/teaching/13-MachineLearning/>

14. Literatur:

[1] *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* by Trevor Hastie, Robert Tibshirani and Jerome Friedman. Springer, Second Edition, 2009.
 full online version available: <http://www-stat.stanford.edu/~tibs/ElemStatLearn/>
 (recommended: read introductory chapter)

[2] *Pattern Recognition and Machine Learning* by Bishop, C. M.. Springer 2006.
 online: <http://research.microsoft.com/en-us/um/people/cmbishop/prml/>
 (especially chapter 8, which is fully online)

15. Lehrveranstaltungen und -formen:
- 294701 Lecture Machine Learning
 - 294702 Exercise Machine Learning

16. Abschätzung Arbeitsaufwand:

Presence time: 42 hours
 Self study: 138 hours
 Sum: 180 hours

17. Prüfungsnummer/n und -name:
- 29471 Machine Learning (PL), schriftlich, eventuell mündlich, 180 Min., Gewichtung: 1.0
 - V Vorleistung (USL-V), schriftlich, eventuell mündlich

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Parallele und Verteilte Systeme

Modul: 42220 Marketing I

2. Modulkürzel:	100160111	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dominik Hettich		
9. Dozenten:	Torsten Bornemann		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Diese Veranstaltung vertieft die institutionelle Perspektive des Marketing. Studierende erlangen darin besondere Kenntnisse zum Marketing von Business-to-Business- bzw. Dienstleistungsunternehmen. Insbesondere sind Studierende mit Abschluss der Veranstaltung in der Lage, Marketingstrategien, -konzepte und -instrumente auf die spezifischen institutionellen Rahmenbedingungen des Business-to-Business- bzw. Dienstleistungskontext anzuwenden.		
13. Inhalt:	Grundlegende Aspekte des B2B-Marketing; Organisationales Kaufverhalten; Besonderheiten des Marketingmix im B2B-Bereich; Grundlagen des Dienstleistungsmarketing; Dienstleistungsqualität; Marketingstrategische Besonderheiten von Dienstleistungen; Instrumentelle Besonderheiten des Dienstleistungsmarketing; Vorlesungsvorträge von Firmenexperten.		
14. Literatur:	<ul style="list-style-type: none"> • Vorlesungsskript und Übungsunterlagen • Backhaus, K., Voeth, M. (2009), Industriegütermarketing, 9. Aufl., München. • Homburg, Christian (2012), Marketingmanagement, 4. Aufl., Wiesbaden. • Zeithaml, V.A., Bitner, M.J. (2003), Services Marketing: Integrating Customer Focus Across the Firm, 3. Aufl., McGraw-Hill. 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 422201 Vorlesung Business-to-Business- und Dienstleistungsmarketing • 422202 Übung Business-to-Business- und Dienstleistungsmarketing 		
16. Abschätzung Arbeitsaufwand:	<u>Vorlesung</u> Präsenzzeit: 28 h Selbststudium: 62 h <u>Übung</u> Präsenzzeit: 28 h Selbststudium: 62 h		

Gesamtstundenanzahl: 180 h

17. Prüfungsnummer/n und -name:	42221 Marketing I (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0
18. Grundlage für ... :	42230 Marketing II
19. Medienform:	
20. Angeboten von:	ABWL und Marketing

Modul: 29640 Mikrocontroller

2. Modulkürzel:	051230115	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>Knowledge of at least one programming language and knowledge in the field of computer science or similar subjects.</p> <p>Kenntnisse in mindestens einer Programmiersprache und in mindestens einem Fach aus dem Bereich dem Bereich der Technischen Informatik oder ähnlichen Fächern.</p>		
12. Lernziele:	<p>Students are able to master the practical programming of microcontrollers and are familiar with classical architectures.</p> <p>Historical Overview Microcontroller architectures Applications of microcontrollers Instruction set classic microcontroller Assembly language programming of microcontrollers C programming for microcontrollers</p> <p>Studierende beherrschen die praktische Programmierung von Mikrocontrollern und kennen klassische Architekturen.</p> <ul style="list-style-type: none"> • Historische Übersicht • Mikrocontroller-Architekturen • Einsatzgebiete von Mikrocontrollern • Befehlssatz klassischer Microcontroller • Assembler-Programmierung von Mikrocontrollern • C-Programmierung von Mikrocontrollern 		
13. Inhalt:	<p>Microcontrollers (also called μController, μC, MCU) are IC's that combine at least peripheral functions on a single chip. In many cases, the working and programming memory is also partially or completely on the same chip . A microcontroller is practically a one-chip computer system.</p>		

The number of built-in microcontroller exceeds by far the number of microprocessors . A microcontroller is often part of an embedded system in devices of everyday life like washing machines, smart cards (money, telephone cards), consumer electronics (VCRs, disc players, radios, televisions, remote controls), office electronics, motor vehicles (ECU for ABS, airbag, engine, instrument cluster, ESP, etc.), mobile phones and even in clocks and watches. In addition they are found on virtually all computer peripherals including keyboards, mouse, printers, monitors, scanners etc.

Microcontrollers are adapted to performance and respective features of the application. Therefore they have significant advantages in cost and power consumption compared with normal computers. Small microcontrollers are available in high numbers for less than 1\$.

Als Microcontroller (auch μ Controller, μ C, MCU) werden ICs bezeichnet, die mit dem Prozessor mindestens Peripheriefunktionen auf einem Chip vereinen. In vielen Fällen befindet sich der Arbeits- und Programmspeicher ebenfalls teilweise oder komplett auf dem gleichen Chip. Ein Mikrocontroller ist praktisch ein Ein-Chip-Computersystem. Die Anzahl der verbauten Mikrocontroller überschreitet bei weitem die Zahl der Mikroprozessoren.

Der Mikrocontroller tritt in Gestalt von eingebetteten Systemen im Alltag oft unbemerkt in technischen Gebrauchsartikeln auf, zum Beispiel in Waschmaschinen, Chipkarten (Geld-, Telefonkarten), Unterhaltungselektronik (Videorekordern, CD-/DVD-Playern, Radios, Fernsehgeräten, Fernbedienungen), Büroelektronik, Kraftfahrzeugen (Steuergeräte für z.B. ABS, Airbag, Motor, Kombiinstrument, ESP usw.), Mobiltelefonen und sogar in Uhren und Armbanduhren. Darüber hinaus sind sie in praktisch allen Computer-Peripheriegeräten enthalten (Tastatur, Maus, Drucker, Monitor, Scanner uvm.).

Mikrocontroller sind in Leistung und Ausstattung auf die jeweilige Anwendung angepasst. Daher haben sie gegenüber normalen Computern deutliche Vorteile bei den Kosten und der Leistungsaufnahme. Kleine Mikrocontroller sind in höheren Stückzahlen für deutlich unter 1\$, - verfügbar.

Aus <http://de.wikipedia.org/wiki/Mikrocontroller>

14. Literatur:	<ul style="list-style-type: none"> • Jörg Wiegelmann, Softwareentwicklung in C für Mikroprozessoren und Mikrocontroller: C- Programmierung für Embedded-Systeme, 2009 <p>More literature is named in the lecture</p>
15. Lehrveranstaltungen und -formen:	296401 Vorlesung mit Übung Mikrocontroller
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden</p> <p>Selbststudium: 138 Stunden</p> <p>Gesamt: 180 Stunden</p>

17. Prüfungsnummer/n und -name: 29641 Mikrocontroller (PL), schriftlich oder mündlich, 90 Min.,
Gewichtung: 1.0, Schriftliche Prüfung von 120 Min. oder
mündlichen Prüfung von 30 Min.

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 29720 Mobile Computing

2. Modulkürzel:	051200166	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> • Kurt Rothermel • Frank Dürr 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Rechnernetze		
12. Lernziele:	<p>The knowledge that has been acquired in the course "Computer Networks I" regarding concepts, protocols, and technologies of computer networks , will be extended to mobile devices and wireless communication systems and procedures. The objective of this lecture is to understand problems that might occur in the usage of mobile devices as well as to obtain knowledge to develop solutions for these problems and to communicate with experts. The Participants will learn about advantages and the disadvantages of specific wireless communication technologies for mobile devices and will be able to use appropriate protocols for the applications or modify them as needed. The exercises are used to provide practical experience in programming, analysis, performance evaluation of mobile and wireless communication systems as well as the expertise in the usage of appropriate tools.</p>		
13. Inhalt:	<ol style="list-style-type: none"> 1. Fundamentals of wireless data transmission 2. Media access for wireless networks 3. Location Management 4. Wireless Wide Area Networks 5. Wireless networks (local/personal) 6. Ad-hoc Networks: Exchange, Location administration 7. Mobility in IP-networks 8. Transport layer protocols for mobile systems 9. Location of services 10. Mobile data access 11. Introduction 12. Wireless data transmission 13. Location Management 14. Wireless 15. Telephone communication systems : GSM, GPRS,UMTS 16. Wireless networks (local/personal): 802.11, Bluetooth 		

	17. Ad-hoc Networks: Routing, Location Management 18. Internetworking: Mobile IP, Cellular IP 19. Transport layers for mobile systems 20. Location of services : Problem, JINI, UpnP 21. Mobile data access: Broadcast Scheduling, Hoarding
14. Literatur:	Charles E. Perkins: Mobile IP: Design Principles and Practices. 1997 James D. Solomon: Mobile IP: The Internet Unplugged. 1998 Jochen Schiller: Mobile Communications. 2000 Jörg Roth: Mobile Computing: Grundlagen, Technik und Konzepte. 2002 Kian-Lee Tan, Beng-Chin Ooi: Data Dissemination in Wireless Computing Envi-ronments. 2000 Tomasz Imielinski, Henry F. Korth (ed.): Mobile Computing. 1996
15. Lehrveranstaltungen und -formen:	297201 Vorlesung mit Übung Mobile Computing
16. Abschätzung Arbeitsaufwand:	Mobile Computing Vorlesung - Präsenzzeit: 21 Stunden - Selbststudium: 69 Stunden Mobile Computing Übungen - Präsenzzeit: 21 Stunden - Selbststudium: 69 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29721 Mobile Computing (PL), schriftlich oder mündlich, Gewichtung: 1.0, Prüfungsdauer: 90 min schriftlich oder 30 min mündlich Exam duration: 90 min wriitten exam or 30 min oral exam • V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	Folien, Tafel
20. Angeboten von:	Verteilte Systeme

Modul: 10120 Modellbildung und Simulation

2. Modulkürzel:	051240010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dirk Pflüger		
9. Dozenten:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 10190 Mathematik für Informatiker und Softwaretechniker • Modul 10240 Numerische und Stochastische Grundlagen der Informatik 		
12. Lernziele:	Beherrschung des grundsätzlichen Vorgehens in der Modellbildung. Kenntnis einer Auswahl diskreter und kontinuierlicher Modelle und entsprechender Simulationsmethoden. Fähigkeit, mit den erlernten Kenntnissen selbständig numerische Methoden problemorientiert um- und einzusetzen.		
13. 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. Den Hauptteil der Vorlesung bilden hierbei diskrete Modelle sowie deren Behandlung, aber auch kontinuierliche Modelle werden ergänzend gestreift. Ob diskrete Ereignissimulation, spieltheoretische Ansätze, Zelluläre Automaten, Räuber-Beute Modelle oder Fuzzy-Mengen: die verschiedenen Modellierungsansätze sind so vielfältig wie die Problemstellungen, auf die sie angewendet werden. Verkehrssimulation, Populationswachstum, Wahlen oder Regelung sind nur einige der Anwendungsbereiche aus den Natur- und Ingenieurwissenschaften.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Modellbildung und Simulation - Eine anwendungsorientierte Einführung; Bungartz, H.-J., Zimmer, S., Buchholz, M., Pflüger, D., Springer Verlag, eXamen.press, 2013, ISBN 978-3-642-38656-6 		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 101201 Vorlesung Modellbildung und Simulation• 101202 Übung Modellbildung und Simulation
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	10121 Modellbildung und Simulation (PL), schriftlich, eventuell mündlich, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Simulation großer Systeme

Modul: 29730 Modelling, Simulation, and Specification

2. Modulkürzel:	051711020	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Master-level understanding of and practical experience with fundamental models of computation and their simulation, ability to apply them to embedded systems specification.		
13. Inhalt:	<p>Given the complexity and implementation cost of contemporary electronic systems, it is essential to specify their intended functionality before elaborating the implementation. This course focuses on the model-based and executable specification of embedded systems and covers the following topics:</p> <ul style="list-style-type: none"> • Hierarchical concurrent state machine models; • Kahn process networks, synchronous data flow networks; • Specification of timing, concurrency, and non-functional aspects; • Object-oriented modelling of embedded systems; • Event-driven simulation; • Modelling levels with emphasis on transaction level modelling; • Application to embedded systems specification; • SystemC. 		
14. Literatur:	<ul style="list-style-type: none"> • Lecture Notes "Modelling, Simulation, and Specification". • Jantsch: Modeling Embedded Systems and SoCs Concurrency and Time in Models of Computation. Morgan Kaufman Publishers, 2004. • Black, D.; Donovan, D.: SystemC from the Ground Up. Kluwer Academic Publishers, 2004. 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 297301 Vorlesung Modelling, Simulation, and Specification • 297302 Übung Modelling, Simulation, and Specification 		

Modul: 55650 Multimodal Interaction for Ubiquitous Computers

2. Modulkürzel:	051900033	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Albrecht Schmidt		
9. Dozenten:	<ul style="list-style-type: none"> • Albrecht Schmidt • Niels Henze 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Basics of human computer interaction		
12. Lernziele:	Broad understanding for methods and concepts of multimodal interactions of personal computers, in particular for mobile systems, vehicles, tedious devices and environments.		
13. Inhalt:	<ul style="list-style-type: none"> • Interaction with mobile phones • User interfaces for vehicles • Interaction with intelligent environments • Interactive interfaces and gestures • Tangible user interfaces • Speech input and output • Camera-based interaction • Physiological sensors as interfaces between human and computer • Activities, context and emotions as input • Methods and techniques for designing user interfaces • Approaches for evaluating user interfaces 		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 556501 Lecture Multimodal Interaction for Ubiquitous Computers • 556502 Exercise Multimodal Interaction for Ubiquitous Computers 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	55651 Multimodal Interaction for Ubiquitous Computers (PL), schriftliche Prüfung, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme		

Modul: 42460 Numerische Simulation

2. Modulkürzel:	051240060	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:	<ul style="list-style-type: none"> • Dirk Pflüger • Stefan Zimmer • Miriam Mehl 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISG →</p> <p>BA (Komb) Informatik, PO 2009, 5. Semester → Module im Nebenfach -->Katalog ISW →</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>080300100 Mathematik für Informatiker und Softwaretechniker und 051240005 Numerische und Stochastische Grundlagen der Informatik bzw. 051240006 Einführung in die Numerik und Stochastik für Softwaretechniker 051240020 Grundlagen des wissenschaftlichen Rechnens</p>		
12. Lernziele:	Fähigkeit zur Implementierung numerischer Methoden und Entwicklung und Umsetzung geeigneter Datenstrukturen.		
13. Inhalt:	Strukturmechanik, Strömungsmechanik, Finite Elemente, Finite Differenzen sowie praktische Aspekte der effizienten und parallelen Umsetzung auf Rechnern.		
14. Literatur:	<ul style="list-style-type: none"> • Griebel, Dornseifer, Neunhoeffler: Numerical simulation in fluid dynamics : a practical introduction; SIAM, 1998 / Numerische Simulation in der Strömungsmechanik; Vieweg 1995 • Griebel, Knapek, Zumbusch, Caglar: Numerische Simulation in der Moleküldynamik : Numerik, Algorithmen, Parallelisierung, Anwendungen; Springer 2004 • Braess: Finite Elemente : Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie; Springer, 2007 		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 424601 Vorlesung Numerische Simulation• 424602 Übung Numerische Simulation
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudiumszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	42461 Numerische Simulation (LBP), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Simulation großer Systeme

Modul: 40680 Optimization

2. Modulkürzel:	051200113	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Marc Toussaint		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Solid basic knowledge in linear algebra and analysis. Basic programming skills.		
12. Lernziele:	Students will learn to identify, mathematically formalize, and derive algorithmic solutions to optimization problems as they occur in nearly all disciplines, e.g. Machine Learning, Combinatorial Optimization, Computer Vision, Robotics, Simulation. The focus will be on continuous optimization problems (including as they arise from relaxations of discrete problems), including convex problems, quadratic & linear programming, but also non-linear black-box problems. The goal is to give an overview of the various approaches and mathematical formulations and practical experience with the basic paradigms.		
13. Inhalt:	<p>Optimization is one of the most fundamental tools of modern sciences. Many phenomena -- be it in computer science, artificial intelligence, logistics, physics, finance, or even psychology and neuroscience -- are typically described in terms of optimality principles. The reason is that it is often easier to describe or design an optimality principle or cost function rather than the system itself. However, if systems are described in terms of optimality principles, the computational problem of optimization becomes central to all these sciences.</p> <p>This lecture aims give an overview and introduction to various approaches to optimization together with practical experience in the exercises. The focus will be on continuous optimization problems and we will cover methods ranging from standard convex optimization and gradient methods to non-linear black box problems (evolutionary algorithms) and optimal global optimization. Students will learn to identify, mathematically formalize, and derive algorithmic solutions to optimization problems as they occur in nearly all disciplines. A preliminary list of topics is:</p> <ul style="list-style-type: none"> • gradient methods, log-barrier, conjugate gradients, Rprop • constraints, KKT, primal/dual • Linear Programming, simplex algorithm • (sequential) Quadratic Programming • Markov Chain Monte Carlo methods • 2nd order methods, (Gauss-)Newton, (L)BFGS • blackbox stochastic search, including a discussion of evolutionary algorithms 		

Please also refer to the course web page: <http://ipvs.informatik.uni-stuttgart.de/mlr/marc/teaching/13-Optimization/>

14. Literatur:

15. Lehrveranstaltungen und -formen: 406801 Vorlesung mit Übungen Optimization

16. Abschätzung Arbeitsaufwand: Presence time: 42 hours
Self study: 138 hours
Sum: 180 hours

17. Prüfungsnummer/n und -name: 40681 Optimization (PL), schriftlich, eventuell mündlich, 120 Min.,
Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Parallele und Verteilte Systeme

Modul: 56790 Parallele Numerik

2. Modulkürzel:	051240080	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. Miriam Mehl		
9. Dozenten:	<ul style="list-style-type: none"> • Miriam Mehl • Dirk Pflüger • Stefan Zimmer 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Modul 41590 Einführung in die Numerik und Stochastik für Softwaretechniker oder • Modul 10240 Numerische und Stochastische Grundlagen 		
12. Lernziele:	<p>Die Studenten kennen die wesentlichen parallelisierbaren Algorithmen für zentrale numerische Problemstellungen. Sie erkennen Parallelisierungshindernisse in bekannten und neuen numerischen Algorithmen, können die zu erwartende Skalierbarkeit abschätzen und sind in der Lage, Algorithmen so zu modifizieren, dass die parallele Effizienz erhöht wird ohne wichtige numerische Eigenschaften wie Stabilität und Komplexität zu verlieren.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • parallele Matrix- und Vektoroperationen • parallele Fouriertransformation • parallele QR Zerlegung und Least Squares Probleme • parallele iterative Gleichungssystemlöser • parallele Eigenwert- und Eigenvektorberechnung • parallele Zeitschrittverfahren • parallele Algorithmen für Teilchenwechselwirkungen 		
14. Literatur:	<ul style="list-style-type: none"> • Introduction to High Performance Scientific Computing (Eijkhout, Chow, van de Geijn) (download at http://www.lulu.com/shop/victor-eijkhout/introduction-to-high-performance-scientific-computing/paperback/product-21431780.html;jsessionid=CF30CC0B65B0F349BFBD206D406F8) • Numerical Linear Algebra for High-Performance Computers (Dongarra, Duff, Sorensen, van der Vorst) • Parallel Algorithms for Matrix Computations (Gallivan, Heath, Ng, Ortega,...) • A User's Guide to MPI (Pacheco) • Iterative Methods for Sparse Linear Systems (Saad) 		

- Loesung linearer Gleichungssysteme auf Parallelrechnern (Frommer)
- M. Griebel, S. Knapek, G. Zumbusch, and A. Caglar. Numerische Simulation in der Molekulardynamik. Springer, 2004.
- D. Frenkel and B. Smith. Understanding Molecular Simulation from Algorithms to Applications. Academic Press (2nd ed.), 2002.

15. Lehrveranstaltungen und -formen:

- 567901 Vorlesung Parallele Numerik
- 567902 Übung Parallele Numerik

16. Abschätzung Arbeitsaufwand:

17. Prüfungsnummer/n und -name:

56791 Parallele Numerik (PL), schriftliche Prüfung, 90 Min.,
Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 29650 Parallele Programmierung

2. Modulkürzel:	051230130	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Kenntnisse in mindestens einer Programmiersprache.Kenntnisse in mindestens einem Fach der Technischen Informatik odereinem ähnlichen Fach.		
12. Lernziele:	Studierende beherrschen die Programmierung von Multi-Core Prozessoren und paralleler Rechner bzw. Computing-Systemen.		
13. Inhalt:	<ul style="list-style-type: none"> • Grundlegende Parallelisierungsansätze: Parallelisierung durch Datenzerlegung, parallele lineare Algebra, etc. • Message Passing Interface • Open MP • C-Programmierung für FPGAs • Graphische Programmierung • GPU-Programmierung 		
14. Literatur:	<ul style="list-style-type: none"> • Thomas Rauber und Gundula Rürger, Multicore: Parallele Programmierung (Informatik Im Fokus), 2007 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	296501 Vorlesung mit Übung Parallele Programmierung		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	29651 Parallele Programmierung (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 10250 Parallele Systeme

2. Modulkürzel:	051200065	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Erfahrungen aus dem Bereich Technische Informatik		
12. Lernziele:	Grundlegende Kenntnisse im Bereich paralleler Systeme, z.B. Multi-Core CPUs und deren Programmierung.		
13. Inhalt:	<ul style="list-style-type: none"> • Die Entwicklung vom klassischen Mikroprozessor zur Multi-Core CPU Programmierung paralleler Rechnersysteme • Systolische Arrays, massiv parallele Systeme • Parallele Systeme aus verschiedenen Anwendungsdomänen: ausgewählte Fallbeispiele 		
14. Literatur:	Wird in der Lehrveranstaltung bekannt gegeben.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102501 Vorlesung Parallele Systeme • 102502 Übung Parallele Systeme 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden Gesamt: 180 Stunden		
17. Prüfungsnummer/n und -name:	10251 Parallele Systeme (LBP), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 14740 Partielle Differentialgleichungen (Modellierung, Analysis, Simulation)

2. Modulkürzel:	080300006	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	unregelmäßig
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Christian Rohde		
9. Dozenten:	<ul style="list-style-type: none"> • Barbara Wohlmuth • Christian Rohde • Barbara Kaltenbacher 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:	<p><i>Zulassungsvoraussetzung: Orientierungsprüfung</i></p> <p><i>Inhaltliche Voraussetzung: Höhere Analysis, Numerische Mathematik 2</i></p>		
12. Lernziele:	<ul style="list-style-type: none"> • Grundlagen zur Behandlung von partiellen Differentialgleichungen. • Erwerb von vertieften Fähigkeiten in einem modernen Teilgebiet der Analysis bzw. Numerik, die als Grundlage des Verständnisses aktueller Forschungsthemen dienen. 		
13. Inhalt:	<p>Modellierung:</p> <ul style="list-style-type: none"> • Herleitung elementarer Typen aus Anwendungen. <p>Analysis:</p> <ul style="list-style-type: none"> • Klassifizierung linearer partieller Differentialgleichungen, elementare Lösungstechniken (Fundamentallösungen, Wellen,...), klassische Existenztheorie in Hölderräumen, schwache Existenztheorie in Sobolevräumen, Asymptotik und qualitatives Verhalten. <p>Numerik:</p> <ul style="list-style-type: none"> • Finite-Differenzen Verfahren, Finite-Elemente Verfahren, effiziente Gleichungslöser. Datenstrukturen,Gittererzeugung. 		
14. Literatur:	<i>Wird in der Vorlesung bekannt gegeben.</i>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 147401 Vorlesung Partielle Differentialgleichungen • 147402 Übungen zur Vorlesung Partielle Differentialgleichungen 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	63h	
	Selbststudium/Nacharbeitszeit:	187h	
	Prüfungsvorbereitung:	20h	
	Gesamt:	270h	

17. Prüfungsnummer/n und -name: 14741 Partielle Differentialgleichungen (Modellierung, Analysis, Simulation) (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0, Prüfungsvorleistung: Übungsschein

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 48570 Practical Course Visual Computing

2. Modulkürzel:	051900111	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Thomas Ertl		
9. Dozenten:	Thomas Ertl		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Basics of Computer Graphics		
12. Lernziele:	During this practical course, students will learn about approaches to rendering and visual computing technologies and will know how to implement these. They will learn about polygon based approach as well as volume rendering approaches. The students will learn, how to proceed a small project on their own (independently).		
13. Inhalt:	OpenGLQt-FrameworkRaytracingVolume RenderingIndependent Project		
14. Literatur:	<ul style="list-style-type: none"> • OpenGL Programming Guide - Third Edition (OpenGL 1.2) , Masonn Woo, Jackie Neider, Tom Davis, Dave Shreiner, Addison Wesley, 1999 • Programming with Qt - First Edition, Matthias Kalle Dalheimer, O'Reilly,1999 • An Introduction to Ray Tracing, Andrew S. Glassner, Academic Press, 1989 • Computer Graphics - Principle and Practice - Second Edition, Foley, van Dam, Feiner, Huges, Addison Wesley, 1990 		
15. Lehrveranstaltungen und -formen:	485701 Lab Practical Course Visual Computing		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	48571 Practical Course Visual Computing (LBP), schriftlich oder mündlich, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29660 Programmanalysen und Compilerbau

2. Modulkürzel:	051510311	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. Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Kenntnisse, die in etwa den Inhalten des Moduls 10150 - Grundlagen des Compilerbaus und der Programmiersprachen - des Bachelor-Studiums entsprechen, sind dringend empfohlen.		
12. Lernziele:	Die Studierenden haben grundlegende Kenntnisse über die typischen in Compilern und anderen statischen Programmanalysen verwandten Verfahren erworben, sowohl in Bezug auf Basisanalysen (Kontroll- und Datenflussanalysen) als auch auf weitergehende, zielgerichteten Analysen wie Zeigeranalysen, Abhängigkeitsanalysen oder Slicing. Speziell lernen sie eine Reihe von Codeoptimierungen im Compiler kennen, aber auch diverse Globalanalysen, wie sie in Werkzeugen zur Fehlersuche, zum Reengineering oder zu Architekturanalysen nötig sind.		
13. Inhalt:	<ul style="list-style-type: none"> • Attributgrammatiken (Wiederholung) • Programmanalysen und -Optimierung (Schwerpunkt) • klassische Optimierungen • Lokale und globale Kontrollflussanalyse • Lokale und globale Datenflussanalysen • Dominatoren, Dominatorgrenzen, Kontrollstrukturanalysen • Zeigeranalysen • Seiteneffekt-Analyse • Datenabhängigkeiten, Konfliktanalysen und Registervergabe • SSA-Form und ihre Berechnung • Code-Erzeugung • Implementierung von OOP • Das Laufzeitsystem • Separate Übersetzung • Slicing • Mustersuchen und Klonerkennung • Begriffsanalyse und ihre Anwendungen 		

Orthogonal zu den jeweiligen Analyseverfahren werden die Verwendungen in Codeoptimierung und in Programmanalysen anderer Werkzeuge des Software Engineering aufgezeigt.

14. Literatur:	<ul style="list-style-type: none">• A. V. Aho, M.S. Lam, R. Sethi, J. D. Ullman: Compilers - Principles, Techniques, and Tools, Addison Wesley Verlag (2007)• Morgan, Robert, Building an Optimizing Compiler, 1998• Muchnick, Steven S., Advanced Compiler Design and Implementation, 1997• Wilhelm, Maurer: Übersetzerbau, 2. Auflage, Springer Verlag (1997)• Uwe Kastens: Übersetzerbau, Oldenbourg Verlag (1990)
15. Lehrveranstaltungen und -formen:	296601 Vorlesung mit Übung Programmanalysen und Compilerbau
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	29661 Programmanalysen und Compilerbau (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Softwaretechnologie

Modul: 51740 Quantencomputing

2. Modulkürzel:	050420210	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Apl. Prof. Ulrich Hertrampf	
9. Dozenten:		Ulrich Hertrampf	
10. Zuordnung zum Curriculum in diesem Studiengang:		<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>	
11. Empfohlene Voraussetzungen:		Grundkenntnisse in Mathematik und Theoretischer Informatik, wie sie in „Mathematik für Informatiker“ und „Theoretische Grundlagen der Informatik“ vermittelt werden.	
12. Lernziele:		Die Studierenden lernen die Grundprinzipien kennen, nach denen der im wesentlichen noch immer hypothetische Quantencomputer arbeiten würde. Sie kennen die grundlegenden Quantenalgorithmen von Deutsch, Jozsa, Simon und anderen, den Zusammenhang mit unitären Matrizen, wichtige Sätze wie das No-Cloning-Theorem. Sie haben den Quanten-Suchalgorithmus von Grover und den Primfaktorzerlegungs-Algorithmus von Shor verstanden.	
13. Inhalt:		Die Vorlesung behandelt zunächst die grundlegenden Techniken des Quanten-Computings: Qubits und Quantenregister, Messungen, Hadamard-Transformation, Quantenschaltkreise. Im zweiten Teil werden wichtige Algorithmen vorgestellt: Grovers Suchalgorithmus, Shors Faktorisierungsalgorithmus. Abschnitte über Teleportation und Quantenkryptographie runden das Thema ab.	
14. Literatur:		Matthias Homeister, „Quantum Computing verstehen“, 2. Auflage, Friedr. Vieweg & Sohn, 2008 Jozef Gruska, „Quantum computing“, McGraw-Hill, 1999.	
15. Lehrveranstaltungen und -formen:		517401 Vorlesung mit Übungen Quantencomputing	
16. Abschätzung Arbeitsaufwand:		Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden	
17. Prüfungsnummer/n und -name:		51741 Quantencomputing (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0	
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29670 Rapid Prototyping

2. Modulkürzel:	051230135	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Erfahrungen in mindestens einer Programmiersprache.		
12. Lernziele:	Die Studierenden beherrschen die schnelle Realisierung von Computing-Systemenausgehend von einer Algorithmen-Implementierung unter Verwendung eines Computer-Algebrasystems.		
13. Inhalt:			
14. Literatur:	<ul style="list-style-type: none"> • James O. Hamblen und Michael D. Furman, Rapid Prototyping of Digital Systems: A Tutorial Approach, 2001 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	296701 Vorlesung mit Übung Rapid Prototyping		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	29671 Rapid Prototyping (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29680 Real-Time Programming

2. Modulkürzel:	051510301	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Significant programming experience (not necessarily in real-time application) is highly advisable. • Knowledge of Ada, C/C++ and Unix is helpful, but not required. 		
12. Lernziele:	Students understand the standard terminology of deadline-driven, safety-critical real-time systems. They understand the issues that differentiate such systems from general software systems, and they know about available solutions, if any.		
13. Inhalt:	<ol style="list-style-type: none"> 1) General requirements and terminology of real-time systems 2) Deterministic execution: avoiding language-, implementation- and hardware-induced non-determinisms; coping with limited resources; storage estimation and management; execution time estimation 3) Fault tolerance: Faults and failure modes, N-version programming, voting, forward and backward recovery 4) Simple scheduling regimes: cyclic executives, deadline guarantees 5) Parallelism and priority scheduling regimes: processes, threads, tasks; run-time kernels; task management; interrupt handling 6) Synchronization and communication: semaphores, critical regions, monitors, protected objects, rendezvous, messaging 7) Control of shared resources 8) Distributed Systems: basic concepts; major issues 		
14. Literatur:	<ul style="list-style-type: none"> • Alan Burns and Andy Wellings, Real-Time Systems and Programming Languages, Addison Wesley, 1997 ... or later editions of the Burns/Wellings-Book, e.g., 4.ed. 2009 • Language reference manuals (C++, Java, Ada) are useful at times. 		
15. Lehrveranstaltungen und -formen:	296801 Vorlesung mit Übung Real-Time Programming		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

17. Prüfungsnummer/n und -name: 29681 Real-Time Programming (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Softwaretechnologie

Modul: 29690 Real-Time Video Processing I

2. Modulkürzel:	051230140	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	This course requires knowledge and experience in (at least) one programming language as well as knowledge of the subject "Technische Informatik" or a similar course		
12. Lernziele:	The Students will gain knowledge in the implementation of algorithms, architectures and exemplary processors for real-time video processing		
13. Inhalt:	<ul style="list-style-type: none"> • Introduction: analog/digital Television • Cameras, Image sensors and their characteristics • Image Filtering, Bayer Filter • Motion Analysis • video compression • video communication • video processing • Parallel architecture, video processors and Implementation of hardware components for real-time video processing algorithms 		
14. Literatur:	<ul style="list-style-type: none"> • Roger Clarke und R. J. Clarke von Academic Press Inc, Digital Compression of Still Images and Video (Signal Processing and Its Applications), 1995 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	296901 Vorlesung mit Übung Real-Time Video Processing I		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	29691 Real-Time Video Processing I (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 29700 Real-Time Video Processing II

2. Modulkürzel:	051230142	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Voraussetzung für Real-Time Video Processing II sind Kenntnisse von Real-Time Video Processing I. Alternativ sind Kenntnisse aus einem Fach der Technischen Informatik oder einem ähnlichen Fach oder Kenntnisse im Bereich der Datenkompression oder der Bildverarbeitung oder der Signalverarbeitung Voraussetzung.		
12. Lernziele:	Die Studierenden können praktisch Real-Time Video Processing Systeme aufbauen.		
13. Inhalt:	<ul style="list-style-type: none"> • Vorstellung der Fallstudie eines Video Processing Systems • Auswahl der Algorithmen des Video Processing Systems • Implementierung und Verifikation der Algorithmen • Architektur-Entwicklung des Video Processing Systems • Performance-Analyse der Achitektur • Implementierung und System-Verifikation 		
14. Literatur:	<ul style="list-style-type: none"> • Roger Clarke und R. J. Clarke von Academic Press Inc, Digital Compression of Still Images and Video (Signal Processing and Its Applications), 1995 • More literature is named in the lecture 		
15. Lehrveranstaltungen und -formen:	297001 Vorlesung mit Übung Real-Time Video Processing II		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	29701 Real-Time Video Processing II (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 45740 Rechnernetze II

2. Modulkürzel:	051200168	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. Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> • Kurt Rothermel • Frank Dürr 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Rechnernetze		
12. Lernziele:	<p>Das in der Veranstaltung Rechnernetze I erworbene Verständnis von Konzepten, Protokollen und Technologien von Rechnernetzen wird weiter vertieft. Der Teilnehmer kennt die Funktionsweise der wichtigsten Dienste der Anwendungsschicht des Schichtenmodells. Er kennt die Konzepte zur Realisierung von Netzen auf Anwendungsebene. Er ist im Stande, diese Dienste und Konzepte bei der Konzeption eigener Anwendungen zu nutzen und ebenso eigene Dienste und Systeme zu entwickeln, um konkrete Probleme zu lösen.</p>		
13. Inhalt:	<p>Vorlesung Höhere Kommunikationskonzepte und -Protokolle:</p> <ol style="list-style-type: none"> 1. Einführung 2. Socket-Schnittstelle 3. Präsentation und Kompression 4. Realzeitkommunikation 5. Elektronische Bezahlsysteme 6. Multicast auf Anwendungsschicht 7. Inhaltsbezogene Netze 8. Geographische Kommunikation <p>Vorlesung Peer-to-Peer-Systeme:</p> <ol style="list-style-type: none"> 1. Definition und Anwendungen von Peer-to-Peer-Systemen 2. Theoretische Netzmodelle 3. Unstrukturierte Peer-to-Peer-Systeme 4. Strukturierte Peer-to-Peer-Systeme 5. Komplexe Suchanfragen 6. Sicherheitsmechanismen für Peer-to-Peer-Systeme 7. Software-Rahmenwerke für Peer-to-Peer-Systeme 		
14. Literatur:			

- J.F. Kurose, K.W. Ross, Computer Networking. 4th Edition, 2007
- L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach. 4th Edition, 2007
- Peter Mahlmann, Christian Schindelhauer, P2P Netzwerke: Algorithmen und Methoden., 2007
- Ralf Steinmetz, Klaus Wehrle, Peer-to-Peer Systems and Applications, 2005A.S. Tanenbaum, Computer Networks, 4th Edition, 2003

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 457401 Vorlesung Höhere Kommunikationskonzepte und -protokolle • 457402 Vorlesung Peer-to-Peer-Systeme
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	45741 Rechnernetze II (PL), schriftlich oder mündlich, Gewichtung: 1.0, Prüfungsdauer: 90 min schriftlich oder 30 min mündlich Exam duration: 90 min written exam or 30 min oral exam
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Verteilte Systeme

Modul: 48580 Reinforcement Learning

2. Modulkürzel:	051200888	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Vien Ngo		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Solid knowledge in linear algebra, probability theory and optimization. Rough knowledge of Artificial Intelligence. Fluency in at least one programming language.		
12. Lernziele:	Students will acquire a deep understanding of Reinforcement Learning methods. Reinforcement Learning addresses the problem of learning optimal behavior (strongly related to optimal control) from data. This course will enable students to apply Reinforcement Learning algorithms in simulated domains and real robotic systems.		
13. Inhalt:	<p>Reinforcement Learning considers how an agent, interacting with a world, can improve or learn optimal behavior based on own experience or teacher demonstration. This branch of Artificial Intelligence and Machine Learning has become increasingly important as a foundation of robust intelligent systems and robotics. Optimal exploration (behavior that optimizes the agent's information gain) is a particularly interesting aspect of Reinforcement Learning. This lecture will introduce to the theory of Reinforcement Learning and then discuss state-of-the-art algorithms in this area.</p> <p>motivation and history Markov Decision Processes and Bellman's optimality principle relations to stochastic optimal control theory basic model-free RL methods (TD-Learning, Q-learning, etc) model-based RL methods theory of optimal exploration (Bayesian RL, R-max) relational RL inverse RL, learning from demonstration and instruction information theoretic formulations of RL modern policy search methods (and applications in robotics)</p>		
14. Literatur:	<ul style="list-style-type: none"> - (Main background) R. Sutton and A. Barto, Reinforcement Learning, 1998. This book is freely available online. - (For robotics application) S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, 2006. - (Hardcore theory) C. Szepesvari, Algorithms for Reinforcement Learning, 2010. Draft version is freely available online. - S. LaValle, Planning Algorithms, 2006. http://planning.cs.uiuc.edu/ 		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 485801 Lecture Reinforcement Learning• 485802 Exercise Reinforcement Learning
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	48581 Reinforcement Learning (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Parallele und Verteilte Systeme

Modul: 36410 Requirements Engineering und Software-Architektur

2. Modulkürzel:	-	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. Lars Grunske		
9. Dozenten:	Lars Grunske		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Grundlagen des Software Engineerings		
12. Lernziele:	Die Teilnehmer haben einen umfassenden Überblick über die verfügbaren Methoden und Techniken zum Requirements Engineering und zur Software-Architektur. Sie haben vertiefte Anwendungserfahrung in ausgewählten Methoden und Techniken.		
13. Inhalt:	<ul style="list-style-type: none"> • Methoden des Requirements Engineerings • Beschreibung und Modellierung von Anforderungen • Analyse und Validierung von Anforderungen • Management von Anforderungen • Modellierung, Erstellung und Analyse von Software-Architekturen • Architekturmuster • Requirements Engineering und Architektur im Entwicklungsprozess 		
14. Literatur:	<ul style="list-style-type: none"> • Robertson, Robertson. Mastering the Requirements Process. Addison-Wesley Professional, 2006 • Sommerville, Sawyer. Requirements Engineering: A Good Practice Guide. John Wiley & Sons, 1997 • Bass, Clements, Kazman. Software Architecture in Practice, 2nd edition. Addison-Wesley Professional, 2003 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 364101 Vorlesung Requirements Engineering und Software-Architektur • 364102 Übung Requirements Engineering und Software-Architektur 		
16. Abschätzung Arbeitsaufwand:	<ul style="list-style-type: none"> • Vorlesung Requirements Engineering und Software-Architektur: Präsenzzeit: 31,5 Stunden, Nachbearbeitungszeit: 103,5 • Übung Requirements Engineering und Software-Architektur: Präsenzzeit: 10,5 Stunden, Nachbearbeitungszeit: 34,5 		
17. Prüfungsnummer/n und -name:	36411 Requirements Engineering und Software-Architektur (PL), schriftlich oder mündlich, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Zuverlässige Softwaresysteme		

Modul: 40630 Ringvorlesung Informatik

2. Modulkürzel:	05190044	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Daniel Weiskopf		
9. Dozenten:	Daniel Weiskopf		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Studierenden haben einen Überblick die Forschungsschwerpunkte im Fachbereich Informatik der Universität Stuttgart erhalten. Sie können die Arbeitsweisen und Anforderungen in den verschiedenen Gebieten einschätzen und sind vorbereitet, sich die Vertiefungslinien in Informatik und anschließend das Forschungsgebiet ihrer Masterarbeit (Informatik) nach Ihren Neigungen und Interessen zu wählen. Sie kennen die Grundzüge des wissenschaftlichen Arbeitens und Publizierens.		
13. Inhalt:	<p>Die Ringvorlesung beginnt in Form einer Blockveranstaltung zu Beginn des Wintersemesters; diese wird von verschiedenen Dozenten der Informatik gehalten. Die Dozenten stellen Ihre Forschungsschwerpunkte vor und geben einen Überblick über ihre Vertiefungslinien, Spezialvorlesungen und Forschungsthemen, die für die spätere Wahl eines Forschungsgebiets der Masterarbeit (Informatik) relevant sind. Der Inhalt vermittelt damit einen Eindruck über die ganze Bandbreite der Informatik, wie sie an der Universität Stuttgart vertreten wird.</p> <p>Im Anschluss an die Blockveranstaltung werden in der wöchentlich stattfindenden Vorlesung die Grundzüge des wissenschaftlichen Arbeitens und Publizierens vermittelt. Zudem wird auf die Organisation des Studiums MSc Informatik eingegangen. Die Themen umfassen im Einzelnen:</p> <ul style="list-style-type: none"> • Aufbau des MSc Informatik in Stuttgart, Prüfungsordnung, Struktur • Überblick über das wissenschaftliche Arbeiten • Methoden des wissenschaftlichen Arbeitens • Publizieren und Reviews • Ethik in der Wissenschaft und gute wissenschaftliche Praxis <p>Bitte schauen Sie sich auch unsere aktuelle Webseite dazu an:</p> <p>http://www.ipvs.uni-stuttgart.de/abteilungen/sgs/lehre/lehrveranstaltungen/vorlesungen/WS1415/ringvorlesung_master_inf</p>		

-
14. Literatur:
- M. Deininger, H. Lichter, J. Ludewig, K. Schneider: Studien-Arbeiten. 5. Auflage, vdf, 2005
 - Prüfungsordnung MSc Informatik, Universität Stuttgart, 2012
 - S. Demeyer: Research Methods in Computer Science, Tutorial, ICSM 2011 Conference, 2011
 - Vorschläge zur Sicherung guter wissenschaftlicher Praxis - Empfehlungen der Kommission "Selbstkontrolle in der Wissenschaft". Deutsche Forschungsgemeinschaft, Wiley-VCH, 1998

15. Lehrveranstaltungen und -formen: 406301 Ringvorlesung Informatik

16. Abschätzung Arbeitsaufwand: Präsenzzeit: 21 Stunden

Selbststudium: 69 Stunden

17. Prüfungsnummer/n und -name: 40631 Ringvorlesung Informatik (USL), schriftlich oder mündlich, Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 48600 Robotics I

2. Modulkürzel:	051200999	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Marc Toussaint		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Solid knowledge in linear algebra, probability theory and optimization. Fluency in at least one programming language.		
12. Lernziele:	Students will acquire the basic methodologies to model, control and navigate robots, including trajectory planning, control of dynamic systems and object manipulation.		
13. Inhalt:	<p>The lecture will give an introduction to robotics, focusing on essential theoretical foundations of planning and controlling motion, state estimation and eventually object manipulation. Exercises in simulations and on a real robot are a core element of this lecture to gain practical experience.</p> <ul style="list-style-type: none"> • motivation and history • (inverse) kinematics • path finding and trajectory optimization • (non-)holonomic systems • mobile robots • sensor processing (vision, range sensors) • simulation of robots and environments • object grasping and manipulation 		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 486001 Lecture Robotics I • 486002 Exercise Robotics I 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	48601 Robotics I (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von:

Institut für Parallele und Verteilte Systeme

Modul: 48610 Robotics II

2. Modulkürzel:	051200888	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Vien Ngo		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Course Robotics I		
12. Lernziele:	Students will acquire indepth knowledge of advanced theoretical topics in robotics as well as the state-of-the-art in autonomous robotics, in particular object manipulation, application of Machine Learning in robotics and control theory on modern (compliant) actuators.		
13. Inhalt:	<p>This course combines the foundations of Reinforcement Learning with robotics and control theory and explores in depth advanced topics at the state-of-the-art in autonomous robotics. The course will focus on core topics such as analytical dynamics, stochastic control theory, and machine learning approaches to data-driven robotics. At the end of the course you will be equipped to read and understand relevant research papers to develop beyond this material on your own.</p> <p>Topics:</p> <ul style="list-style-type: none"> - Analytical dynamics (Lagrange, Hamilton, Gauss formulations; contact analysis) - Stochastic optimal control (focus on nonlinear systems) - Inverse optimal control (maximum margin and maximum entropy) - Imitation learning (inverse reinforcement learning) - Policy search (model based and model free) - Model learning (forward and inverse models) 		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 486101 Lecture Robotics II • 486102 Exercise Robotics II 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	48611 Robotics II (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von:

Modul: 48620 Scientific Visualization

2. Modulkürzel:	051900777	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Thomas Ertl		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Filip Sadlo • Daniel Weiskopf 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	<p>Basic concepts of Human Computer Interaction</p> <p>Basic concepts of Computer Graphics</p>		
12. Lernziele:	<p>Student gains expertise about fundamental concepts and techniques of scientific visualization. This includes algorithms and mathematical background, data structures and implementation aspects as well as practical experience with widely available visualization tools.</p>		
13. Inhalt:	<p>Visualization discusses all aspects of visual representations of data gained from experiments, simulations, medical scanning machines, data bases an the like. The aim of visualization is to gain further insights into the data or the generate "simple" representations of complex phenomena or issues. For that, known techniques from the research area of interactive computer graphics as well as novel techniques are applied.</p> <p>The following topics will be discussed:</p> <ul style="list-style-type: none"> • Introduction, history, visualization pipeline • Data aquisition and representation (sampling, reconstruction, grids, data structures) • PerceptionBasic concepts of visual mappings • Visualization of scalar fields (extraction of iso-surfaces, volume rendering) • Visualization of vector fields (particle tracking, texture-based methods, topology) • Tensor fields, multivariate data • Highdimensional data and information visualization 		
14. Literatur:	<ul style="list-style-type: none"> • C. D. Hansen, C. R. Johnson, The Visualization Handbook, 2005 • C. Ware, Information Visualization: Perception for Design, 2004 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 486201 Lecture Scientific Visualization 		

	• 486202 Exercise Scientific Visualization
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	• 48621 Scientific Visualization (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich oder mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Visualisierung und Interaktive Systeme

Modul: 29510 Service Computing

2. Modulkürzel:	052010004	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Frank Leymann • Dimka Karastoyanova 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>A diversity of technologies enable nowadays computer-based interactions on the Web and on the Internet. The aim of this course is to make the students familiar with some of the most pervasive technologies that come together to form the Web and the Internet as we know it, and that enable to build large-scale application systems.</p>		
13. Inhalt:	<p>At first, we will cover the Web-centric technologies that enable the interaction of humans with Web content, e.g. HTTP, SMTP, AJAX, CSS and MIME . On the server-side part of technology, we will treat several Java EE technologies such as portlets, servlets, and JSP.</p> <p>The second part of the course will cover a set of technologies that are prominent in the landscape of Service-Oriented Architecture (SOA). In a nutshell, SOA is a paradigm that advocates the creation of complex, value added applications by reusing and composing independent and loosely coupled (software) services. We will dissect prominent SOA concepts like service discovery, addressing, policies, Service Bus, coordination protocols and service compositions. The architectural concepts will be complemented with an outlook of the technologies that embody them in the landscape of enterprise computing. In particular, we will cover several XML-centric technologies that sit at the core of Web services, e.g. XSD, SOAP, WSDL and Policy. In addition to the SOAP-based approach to Web services, we will also explore their REST aspect. Building on this portfolio of technologies, we will discuss the relationships between Web service technologies and “hot” items on the enterprise computing agenda such as autonomic/organic computing and cloud computing.</p>		
14. Literatur:	<p>S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson: “Web Services Platform Architecture”, Prentice Hall 2005</p>		

G. Alonso, F. Casati, H. Kuno, V. Machiraju: "Web Services", Springer 2004

E. Wilde: "World Wide Web", Springer 1999

M.P. Papazoglou: "Web Services: Principles & Technology", Pearson Education Limited 2008

N.M. Josuttis: "SOA in Practice: The Art of Distributed System Design", O'Reilly 2007

Th. Erl: "SOA: Entwurfsprinzipien für serviceorientierte Architektur", Addison-Wesley 2008

D.A. Chappell: "Enterprise Service Bus", O'Reilly 2004

15. Lehrveranstaltungen und -formen:	295101 Vorlesung mit Übungen Service Computing
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 29511 Service Computing (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0, Wenn Prüfung in diesem Fach, dann K E I N E Prüfung in Services and Service Composition (auch nicht innerhalb der VTL) möglich! • V Vorleistung (USL-V), schriftlich oder mündlich
18. Grundlage für ... :	29520 Ausgewählte Themen des Service Computing
19. Medienform:	Lecture and accompanying exercises
20. Angeboten von:	Architektur von Anwendungssystemen

Modul: 31080 Service Engineering

2. Modulkürzel:	[pord.modulcode]	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Vasilios Andrikopoulos • Frank Leymann 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Service Computing, Lecture and Exercise, 4 SWS or Services and Service Composition, Lecture and Exercise, 4 SWS		
12. Lernziele:	<p>Students attending the course and exercise lectures in this module will become knowledgeable on the complete lifecycle of software services and the related methodologies, techniques, and best practices for the development and operation of services and service-oriented architectures. The students will be capable of addressing software project management concerns related to service orientation. Hands-on experience on the major technologies for service implementation during the practical exercises will allow students to grasp the various aspects of service engineering better. The course combines industrial-led initiatives and standards with rigorous academic research results and provides students with an up-to-date picture of the state of the art in service engineering.</p>		
13. Inhalt:	<p>This module spans the lifecycle of software services and discusses methodologies, techniques, best practices and open issues concerning the development and operation of services and service-oriented architectures (SOAs). Software project management concerns related to service orientation are also discussed as part of this course. Presentations of relevant and dominant technologies for service implementation are also included, but the emphasis is on how and when they can be used for service engineering rather than their technical details. The course combines industrial-led initiatives and standards with rigorous academic research results to provide an up-to-date picture of the state of the art in service engineering.</p> <p>During the course the following topics are discussed:</p> <ul style="list-style-type: none"> - Services Lifecycle - SOA Analysis & Design - SOA Design Principles & Patterns 		

- Model-Driven Service Development
- Realizing Web Services
- Designing and Implementing RESTful Services
- Service Composition and Mashups
- Testing
- SOA Project Management
- Service Governance
- Software, Service and Cloud Engineering

14. Literatur:	For each course and exercise lecture a list of relevant material in books, academic papers and online resources is provided with the lecture slides.
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 310801 Vorlesung Service Engineering• 310802 Übung ServLab
16. Abschätzung Arbeitsaufwand:	<ul style="list-style-type: none">- Präsenzzeit: 42 Stunden- Selbststudiumszeit: 138 Stunden
17. Prüfungsnummer/n und -name:	31081 Service Engineering (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Architektur von Anwendungssystemen

Modul: 46660 Service Management and Cloud Computing, and Evaluation

2. Modulkürzel:	052000111	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Kristof Klöckner		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Service Computing, Business Process Management		
12. Lernziele:	The students will learn the basics of systems management and cloud computing.		
13. Inhalt:	<p>Cloud Computing is an emerging paradigm for consumption and delivery of IT based services, based on concepts derived from consumer internet services, like self-service, apparently unlimited or elastic resources and flexible sourcing options. In this course we will discuss the technical foundations of cloud computing, as well as the business models associated with it.</p> <p>We will start by looking at virtualization and service management as the technical underpinnings. We will then look at infrastructure services and platform services, with a particular focus on emerging programming models for the cloud. We will discuss the trade-offs made between consistency and availability as well as extensions to "traditional" programming models. We also look at the life-cycle of applications in the cloud.</p> <p>Finally, we will look some of the challenges of Software as a Service, like multi-tenancy.</p> <p>Throughout the course, we will look both at existing products and services as well as the theoretical underpinnings.</p> <p>The course will be held as a combination of lectures and participant discussion.</p>		
14. Literatur:	To be announced in the lecture.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 466601 Vorlesung Service Management and Cloud Computing, and Evaluation • 466602 Excercise Service Management and Cloud Computing, and Evaluation 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		

Modul: 42520 Services and Service Composition

2. Modulkürzel:	052010008	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Dimka Karastoyanova		
9. Dozenten:	<ul style="list-style-type: none"> • Vasilios Andrikopoulos • Dimka Karastoyanova • Frank Leymann 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>The students will learn the foundations of the SOA and REST Architectural styles and technologies that can be used for their realization. The concept of service and the principle of loose coupling will be clarified. The students will be able to realize Service based applications using the Web Service technology. The students will be knowledgeable of the concepts workflow, service composition and how to apply them using workflow languages in order to create complex, value-added applications.</p>		
13. Inhalt:	<p>Architectural styles: SOA and REST Basic principles: loose coupling vs. tight coupling Service Technologies (WSDL, Policy, WS-Addressing, SOAP) Virtualization and Middleware (Service Bus, etc.) Basics of the Workflow Technology Business Process Re-engineering Workflow Life Cycle Workflow Management System Architecture Workflow Languages (FDL, BPEL)</p>		
14. Literatur:	<ul style="list-style-type: none"> • F. Leymann, D. Roller, Production Workflow, 2000 • S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005 • W. van der Aalst, K. van Hee, Workflow Management, 2002 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 425201 Vorlesung Services and Service Compositions • 425202 Übung Services and Service Compositions 		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 48 Stunden Selbststudiumszeit: 132 Stunden</p>		

17. Prüfungsnummer/n und -name: 42521 Services and Service Composition (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0, Wenn Prüfung in diesem Fach, dann **K E I N E** Prüfung in Service Computing und/oder Business Process Management (auch nicht innerhalb der VTL) möglich!

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Architektur von Anwendungssystemen

Modul: 14480 Sichere und zuverlässige Softwaresysteme

2. Modulkürzel:	051520115	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. Lars Grunske		
9. Dozenten:	Lars Grunske		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Teilnahme an der Lehrveranstaltung "Einführung in die Softwaretechnik" 		
12. Lernziele:	Die Teilnehmer beherrschen Techniken, deren Anwendung die Zuverlässigkeit der Software verbessert oder garantiert. Dabei stehen formale Ansätze im Vordergrund.		
13. Inhalt:	Notationen und Verfahren zur formalen Beschreibung und Prüfung der Software Notationen und Verfahren zum Sicherheits- Performanz- und Zuverlässigkeitsnachweis Verfahren zur Erstellung von sicheren und zuverlässigen Systemen		
14. Literatur:	A. Alessandro Birolini, Reliability Engineering, Springer Verlag Berlin Heidelberg 2010. B. Nancy G. Leveson, Safeware: System Safety and Computers. Addison-Wesley, 1995. C. Nancy G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety. MIT Press, 2011.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 144801 Vorlesung Sichere und zuverlässige Softwaresysteme • 144802 Übung Sichere und zuverlässige Softwaresysteme 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	14481 Sichere und zuverlässige Softwaresysteme (PL), schriftlich oder mündlich, 90 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Zuverlässige Softwaresysteme		

Modul: 16500 Software Engineering

2. Modulkürzel:	051520110	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. Lars Grunske		
9. Dozenten:	Lars Grunske		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 4. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 4. Semester → Ergänzende Spezialisierungsmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Einführung in die Softwaretechnik • Programmentwicklung 		
12. Lernziele:	Die Teilnehmer haben tiefe und umfassende Kenntnisse auf dem Gebiet des Softwareprojekt-Managements und in den Techniken der Software-Bearbeitung.		
13. Inhalt:	Ergänzend zur "Einführung in die Softwaretechnik" und daran anknüpfend behandelt diese Lehrveranstaltung folgende Themen: <ul style="list-style-type: none"> • Softwarequalitätssicherung • Organisationsaspekte der Software-Bearbeitung • Software-Prozesse, Prozess-Bewertung und -Verbesserung • Software-Wartung • Weitere ausgewählte Kapitel des Software Engineerings 		
14. Literatur:	<ul style="list-style-type: none"> • Ludwig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2. Aufl. 2010 • Liggesmeyer P., Software-Qualität. Testen, Analysieren und Verifizieren von Software. Spektrum Akademischer Verlag, 2002. 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 165001 Vorlesung Software Engineering • 165002 Übung Software Engineering 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17. Prüfungsnummer/n und -name:	16501 Software Engineering (PL), schriftliche Prüfung, 90 Min., Gewichtung: 1.0,		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Softwaretechnologie		

Modul: 56550 Software Verification

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. Lars Grunske		
9. Dozenten:	Antonio Filieri		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Theoretical CS and Software Engineering modules (eg. SEfST, SZS)		
12. Lernziele:	The course has the objective to provide theoretical and practical knowledge about the essential software verification techniques. The goal is to enable students to use and apply them successfully to real world code artifacts. In this respects students will understand the required concepts of the different software verification techniques. Of great importance are, beside standard and time verification techniques also probabilistic verification techniques - these will be explained in detail and students will be able to apply them for quality assurance of probabilistic properties like safety, reliability and performance.		
13. Inhalt:	This course will introduce the foundations of this area Process algebraic specification software systems Abstract interpretation and symbolic execution Axiomatic semantics of software systems Software model checking Modeling time in computation and timed verification for software systems Quantitative analysis and probabilistic model checking Probabilistic symbolic execution Statistical methods in software verification		
14. Literatur:	1) Michael Huth and Mark Ryan. Logic in Computer Science: Modelling and Reasoning about Systems, second edition. Cambridge University Press, 2004.		

2) Christel Baier and Joost-Pieter Katoen. Principles of Model Checking.
The MIT Press, 2008

15. Lehrveranstaltungen und -formen:

- 565501 Vorlesung Software Verification
 - 565502 Übung Software Verification
-

16. Abschätzung Arbeitsaufwand:

Präsenzzeit: 42 Stunden

Nachbearbeitungszeit: 138 Stunden

17. Prüfungsnummer/n und -name:

- 56551 Software Verification (PL), schriftlich oder mündlich, 90 Min.,
Gewichtung: 1.0
 - V Vorleistung (USL-V), schriftlich, eventuell mündlich
-

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 42810 Software-Qualitätssicherung und -Wartung

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. Stefan Wagner		
9. Dozenten:	Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Grundlagen des Software Engineerings oder Software Engineering für Softwaretechniker		
12. Lernziele:	Die Teilnehmer verstehen Qualitätssicherungs- und Wartungsprozesse für Software. Sie haben einen umfassenden Überblick über entsprechende Methoden und Techniken, die sie auch einordnen können. Für eine Auswahl davon haben sie vertiefte Anwendungskennnisse.		
13. Inhalt:	<ul style="list-style-type: none"> • Qualitätssicherungs- und Wartungsprozesse • Prozesskontrolle und -steuerung • Qualitätssicherungsmethoden und Qualitätsmodelle • Vorhersagemodelle, Repository Mining, Software Analytics • Programmanalyse und Programmverstehen • Werkzeugunterstützung 		
14. Literatur:	<ul style="list-style-type: none"> • Wagner. Software Product Quality Control. Springer, 2013 • Liggesmeyer. Software-Qualität. Testen, Analysieren und Verifizieren von Software. Spektrum Akademischer Verlag, 2002 • Sneed, Hasitschka, Teichmann. Software-Produktmanagement. Wartung und Weiterentwicklung bestehender Anwendungssysteme. Dpunkt, 2004 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 428101 Vorlesung Software-Qualitätssicherung und -Wartung • 428102 Übung Software-Qualitätssicherung und -Wartung 		
16. Abschätzung Arbeitsaufwand:	<ul style="list-style-type: none"> • Vorlesung: Präsenzzeit: 31,5 Stunden, Nachbearbeitungszeit: 103,5 • Übung: Präsenzzeit: 10,5 Stunden, Nachbearbeitungszeit: 34,5 		
17. Prüfungsnummer/n und -name:	42811 Software-Qualitätssicherung und -Wartung (PL), schriftlich oder mündlich, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Software-Engineering		

Modul: 60140 Sprachbau mit Language Workbenches

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. Stefan Wagner		
9. Dozenten:	Markus Völter		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009 → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012 → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012 → Vorgezogene Master-Module M.Sc. Informatik, PO 2012 → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Compilerbau		
12. Lernziele:	Die Studenten verstehen, warum und wie Modelle für den praktizierenden Softwareentwickler nützlich sind: Kommunikation, Analyse, Synthese. Sie verstehen warum es in vielen Fällen sinnvoll ist, eigene (domänenspezifische) Sprachen zu bauen. Sie verstehen die Mechanismen um Sprachen zu bauen, insbesondere die Fähigkeiten moderner Language Workbenches. Die Studenten sind in der Lage mit JetBrains MPS oder Xtext Sprachen zu bauen.		
13. Inhalt:	Modellierung, Grammatiken, projizierende Editoren, Typsysteme, Codegenerierung, Interpreter. Grundlagen des Sprachdesigns: Ausdrucksfähigkeit vs. Komplexität, Vollständigkeit, Modularisierung, verschiedene Notationen. Wichtige Sprachparadigma, die man in DSLs wiederverwenden kann: imperativ, funktional, zustandsbasiert. Grundlagen der Arbeit mit MPS und/oder Xtext (ggfs. zwei Gruppen).		
14. Literatur:	Buch http://dslbook.org/ + ggfs. diverse wissenschaftliche Papiere		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 601401 Vorlesung Sprachbau • 601402 Übung Sprachbau 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit in Stunden: 56 Selbststudiumszeit in Stunden : 124		
17. Prüfungsnummer/n und -name:	60141 Sprachbau mit Language Workbenches (PL), schriftliche Prüfung, 60 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 48640 Theoretical and Methodological Foundations of Autonomous Systems

2. Modulkürzel:	051200987	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Marc Toussaint		
9. Dozenten:	Marc Toussaint		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 3. Semester → Spezialisierungsmodule -->MINF →		
11. Empfohlene Voraussetzungen:	Solid knowledge in linear algebra, probability theory and optimization. Fluency in at least one programming language.		
12. Lernziele:	Students will acquire a conceptual overview of the challenges and research in intelligent autonomous systems. The course will emphasize the necessity of combining theory with integrated systems, namely the theoretical and computational foundations modeling and solving decision and behavioral problems and the integration in real-world autonomous systems that integrate perception, action and (on-board) computation. The course reflects the conceptual structure of the Major in Autonomous Systems by addressing the methodological foundations of (i) Computational Intelligence and Learning, (ii) Perception and Action, and (iii) System Integration.		
13. Inhalt:	This course discusses the challenges and research in intelligent autonomous systems. It introduces to the basic foundations in the relevant disciplines to enable a holistic view on autonomous systems. This is done using a coherent formalization for concepts which are usually introduced separately. <ul style="list-style-type: none"> • motivation and history • challenges in autonomous systems • frameworks for modeling decision and behavioral problems • computational methods for solving such problems: planning, decision making • system integration • classical Artificial Intelligence and modern probabilistic AI • perception and image processing • learning from data (basic regression and classification) • learning applied in autonomous systems (Reinforcement Learning, adaptive control, system identification) 		
14. Literatur:			

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 486401 Lecture Theoretical and Methodological Foundations of Autonomous Systems• 486402 Exercise Theoretical and Methodological Foundations of Autonomous Systems
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17. Prüfungsnummer/n und -name:	48641 Theoretical and Methodological Foundations of Autonomous Systems (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Parallele und Verteilte Systeme

Modul: 46450 Vertiefungslinie Architektur von Anwendungssystemen

2. Modulkürzel:	[pord.modulcode]	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Lernziele sind bei den einzelnen Modulen, die kombiniert werden, definiert.		
13. Inhalt:	Der Inhalt ist bei den einzelnen Modulen, die kombiniert werden, definiert.		
14. Literatur:	Literatur ist bei den einzelnen Modulen, die kombiniert werden, zu finden.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 464501 Deutschsprachige Lehrveranstaltungen Wintersemester • 464502 Deutschsprachige Lehrveranstaltungen Sommersemester • 464503 courses in english - winter semester • 464504 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:	abhängig von der gewählten Kombination ca: 2 x Präsenzzeit: 42 Stunden 2 x Selbststudium: 138 Stunden		
17. Prüfungsnummer/n und -name:	46451 Vertiefungslinie Architektur von Anwendungssystemen (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0, Keine Kombination von Service Computing und/oder Business Process Management mit Services and Service Composition möglich!		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Architektur von Anwendungssystemen		

Modul: 29330 Vertiefungslinie Datenbanken und Informationssysteme

2. Modulkürzel:	051210555	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Univ.-Prof. Bernhard Mitschang		
9. Dozenten:	<ul style="list-style-type: none"> • Bernhard Mitschang • Melanie Herschel • Holger Schwarz 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:	Vorlesung Modellierung aus dem Bachelor oder gleichwertige Veranstaltungen		
12. Lernziele:	Die Studenten haben vertiefte Kenntnisse auf dem Gebiet der Informationssysteme erworben und können die erlernten Methoden erfolgreich zur Anfertigung wissenschaftlicher Arbeiten sowie zur Lösung von Problemen der Informationsgewinnung, -verarbeitung und -verwaltung anwenden.		
13. Inhalt:	<p>Es müssen 2 Vorlesungen mit Übungen im Gesamtumfang von 8 SWS besucht werden. Dabei kann aus folgenden Veranstaltungen gewählt werden:</p> <ol style="list-style-type: none"> 1) Datenbanken und Informationssysteme (Vorlesung mit Übung, 4 SWS) 2) Advanced Information Management (Vorlesung mit Übung, 4 SWS) 3) Data Warehousing, Data Mining und OLAP-Technologien (Vorlesung mit Übung, 4 SWS) 4) Information Integration (Vorlesung mit Übung, 4 SWS) 		
14. Literatur:	<ul style="list-style-type: none"> • A. Kemper, A. Eickler, Datenbanksysteme - Eine Einführung, 2004 • T. Härder, E. Rahm, Datenbanksysteme - Konzepte und Techniken der Implementierung, 2001 <p>Weitere Literatur wird in den einzelnen Lehrveranstaltungen bekanntgegeben.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 293301 Deutschsprachige Lehrveranstaltungen Wintersemester • 293302 Deutschsprachige Lehrveranstaltungen Sommersemester • 293303 courses in english - winter semester • 293304 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzzeit: 84 Stunden Selbststudium: 276 Stunden</p> <p>Gesamt: 360 Stunden</p>		

17. Prüfungsnummer/n und -name: 29331 Vertiefungslinie Datenbanken und Informationssysteme (PL),
mündliche Prüfung, 45 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Datenbanken und Informationssysteme

Modul: 29340 Vertiefungslinie Intelligent Systems

2. Modulkürzel:	051901555	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Andrés Bruhn		
9. Dozenten:	<ul style="list-style-type: none"> • Andrés Bruhn • Marc Toussaint • Sebastian Pado 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:	<p>Grundlegende Kenntnisse in Mathematik und Bildverarbeitung (z.B. Mathematik für Informatiker und Softwaretechniker 080300100, Imaging Science 051900210) Grundlagen der Künstlichen Intelligenz 10110</p>		
12. Lernziele:	<p>Die Studierenden haben vertiefte Kenntnisse in den Bereichen des Maschinensehens, des maschinellen Lernens, der maschinellen Sprachverarbeitung und der Robotik erworben und können mit den erlernten Methoden wissenschaftliche Arbeiten in diesem Bereich verstehen. Sie haben das notwendige Rüstzeug, um eine Masterarbeit in den zuvor genannten Gebieten anzufertigen.</p> <p>The students have acquired specialised knowledge in the areas of computer vision, machine learning, language processing, and robotics, and they are capable of understanding scientific papers and books from this field. They have the necessary knowledge to begin a Master's thesis in one of the aforementioned areas.</p>		
13. Inhalt:	<p>Es werden Vorlesungen bzw. Vorlesungen mit Übungen im Bereich Intelligente Systeme im Umfang von 8 SWS besucht, die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis tragen und dort inhaltlich beschrieben werden. In dem Vertiefungsmodul Intelligente Systeme gehören hierzu die Veranstaltungen:</p> <ol style="list-style-type: none"> 1) 051900215 Computer Vision (Vorlesung mit Übung, 4 SWS) 2) 051900211 Correspondence Problems in Computer Vision (Vorlesung mit Übung, 4 SWS) 3) 051900205 Grundlagen der künstlichen Intelligenz (Vorlesung mit Übung, 4 SWS) 		

- 4) 051200112 Maschine Learning (Vorlesung mit Übung, 4 SWS)
- 5) 051200113 Optimization (Vorlesung mit Übung, 4 SWS)
- 6) 051200888 Reinforcement Learning (Vorlesung mit Übung, 4 SWS)
- 7) 051200999 Robotics I: Introduction (Vorlesung mit Übung, 4 SWS)
- 8) N/A Advanced Robotics (Vorlesung mit Übung, 4 SWS)
- 9) 052401010 Information Retrieval und Text Mining (Vorlesung mit Übung, 4 SWS)

Students have to attend lectures with and without exercises in the field of Intelligent Systems with a total of 8 SWS. These lectures have to be eligible from the MINF-catalogue (MINF 1-8) where their content is described. The lectures in the specialised module Intelligent Systems include:

- 1) 051900215 Computer Vision (lecture with exercises, 4 SWS)
- 2) 051900211 Correspondence Problems in Computer Vision (lecture with exercises, 4 SWS)
- 3) 051900205 Grundlagen der künstlichen Intelligenz (lecture with exercises, 4 SWS)
- 4) 051200112 Maschine Learning (lecture with exercises, 4 SWS)
- 5) 051200113 Optimization (lecture with exercises, 4 SWS)
- 6) 051200888 Reinforcement Learning (lecture with exercises, 4 SWS)
- 7) 051200999 Robotics I: Introduction (lecture with exercises, 4 SWS)
- 8) N/A Advanced Robotics (lecture with exercises, 4 SWS)
- 9) 052401010 Information Retrieval und Text Mining (lecture with excercises, 4 SWS)

14. Literatur:

15. Lehrveranstaltungen und -formen:
- 293401 Deutschsprachige Lehrveranstaltungen Wintersemester
 - 293402 Deutschsprachige Lehrveranstaltungen Sommersemester
 - 293403 Courses in English - winter semester
 - 293404 Courses in English - summer semester

16. Abschätzung Arbeitsaufwand:
- Präsenzzeit: 84 Stunden
 Selbststudium: 276 Stunden
- Gesamt: 360 Stunden**

17. Prüfungsnummer/n und -name:
- 29341 Vertiefungslinie Intelligent Systems (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0
 - V Vorleistung (USL-V), schriftlich, eventuell mündlich

18. Grundlage für ... :

19. Medienform:

20. Angeboten von:

Modul: 46470 Vertiefungslinie Parallele Systeme

2. Modulkürzel:	-	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Vertiefungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Studenten haben vertiefte Kenntnisse in den Bereichen parallele Systeme sowie in Multi-Core CPUs und deren Programmierung.		
13. Inhalt:	Es werden Vorlesungen bzw. Vorlesungen mit Übungen im Bereich Parallele Systeme im Umfang von (SWS besucht, die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis tragen und dort inhaltlich beschrieben werden. In dem Vertiefungsmodul Parallele Systeme gehören hierzu die Veranstaltungen: 1) Data Compression (Vorlesung mit Übung , 4SWS, 6LP) 2) Parallele Systeme (Vorlesung mit Übung, 4SWS , 6LP)		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 464701 Deutschsprachige Lehrveranstaltungen Wintersemester • 464702 Deutschsprachige Lehrveranstaltungen Sommersemester • 464703 courses in english - winter semester • 464704 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit:	84 Stunden	
	Selbststudium:	276 Stunden	
	Gesamt:	360 Stunden	
17. Prüfungsnummer/n und -name:	46471 Vertiefungslinie Parallele Systeme (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 46440 Vertiefungslinie Programmiersprachen, Compilerbau, Programmanalysen

2. Modulkürzel:	051510350	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule		
11. Empfohlene Voraussetzungen:	Sofern nicht als Teil des Vertiefungsmoduls gewählt, ist das in Modul 57050 Compilerbau vermittelte Wissen Voraussetzung.		
12. Lernziele:	Die Studierenden haben vertiefte Kenntnisse in mehreren Bereichen rund um Programmiersprachen, Übersetzung und Programmanalysen erworben und können mit den erlernten Methoden wissenschaftliche Arbeiten in diesem Bereich verstehen. Sie können Querbezüge zwischen den Bereichen herstellen. Sie haben das notwendige Rüstzeug, um eine Masterarbeit im genannten Umfeld anzufertigen.		
13. Inhalt:	Es werden Vorlesungen bzw. Vorlesungen mit Übungen im Umfang von 8 SWS besucht, die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis für den Vertiefungsmodul tragen. Dazu gehören z. B. die Veranstaltungen zu den Modulen: <ul style="list-style-type: none"> • 57050 Compilerbau • 29660 Programmanalysen und Compilerbau • 29680 Real-Time Programming • 58190 Entwurf und Implementierung eines Compilers Die jeweiligen Inhalte sind bei den Modulen beschrieben. Grundsätzliches Wissen aus dem Modul 57050 wird in jedem Fall vorausgesetzt, auch wenn das Modul nicht als Teil des Vertiefungsmoduls gewählt wird.		
14. Literatur:	ist bei den im Inhalt genannten jeweiligen Modulen angegeben		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 464401 Deutschsprachige Lehrveranstaltungen Wintersemester • 464402 Deutschsprachige Lehrveranstaltungen Sommersemester • 464403 courses in english - winter semester • 464404 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 84 Stunden Selbststudium: 276 Stunden		

17. Prüfungsnummer/n und -name: 46441 Vertiefungslinie Programmiersprachen, Compilerbau, Programmanalysen (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Softwaretechnologie

Modul: 29370 Vertiefungslinie Rechnerarchitekturen und eingebettete Systeme

2. Modulkürzel:	051700555	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Martin Radetzki • Hans-Joachim Wunderlich 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:	Grundlegende Kenntnisse in Rechnerarchitektur und eingebettete Systeme		
12. Lernziele:	Die Studierenden haben vertiefte Kenntnisse in mehreren Bereichen der Technischen Informatik, insbesondere der Rechnerarchitektur und der eingebetteten Systeme erworben und können mit den erlernten Methoden wissenschaftliche Arbeiten in diesem Bereich verstehen. Sie haben das notwendige Rüstzeug, um eine Masterarbeit im Gebiet der Technischen Informatik anzufertigen.		
13. Inhalt:	<p>Es werden Vorlesungen bzw. Vorlesungen mit Übungen aus dem Bereich der Technischen Informatik im Umfang von 8 SWS besucht, die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis tragen und dort inhaltlich beschrieben werden. Zu diesen gehören derzeit:</p> <p>051700020 Hardware Verification and Quality Assessment</p> <p>051710023 Hardware Based Fault Tolerance</p> <p>051711027 Embedded Systems Engineering</p> <p>051711020 Modelling, Simulation and Specification</p>		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 293701 Deutschsprachige Lehrveranstaltungen Wintersemester • 293702 Deutschsprachige Lehrveranstaltungen Sommersemester • 293703 courses in english - winter semester • 293704 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 84 StundenSelbststudium: 276 Stunden		
17. Prüfungsnummer/n und -name:	29371 Vertiefungslinie Rechnerarchitekturen und eingebettete Systeme (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von: Institut für Technische Informatik

Modul: 46460 Vertiefungslinie Sprachverarbeitung

2. Modulkürzel:	-	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Sebastian Pado		
9. Dozenten:	Sebastian Pado		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2012, 6. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik → B.Sc. Informatik, PO 2012, 6. Semester → Vorgezogene Master-Module M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule M.Sc. Informatik, PO 2012, 2. Semester → Vertiefungsmodule		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	Die Studierenden erhalten einen Überblick über die Grundkonzepte, Forschungsfragen, Ressourcen, Methoden und Anwendungen der Computerlinguistik. --- Students become familiar with the main concepts, research questions, resources, methods and applications of Computational Linguistics.		
13. Inhalt:	Dieses Modul setzt sich zusammen aus 1. dem Modul 35150 "Methods in Computational Linguistics" und 2. einem wählbaren Modul aus dem Katalog MCL 6 des Studiengangs M.Sc. Computational Linguistics. --- This module consists of 1. the module 35150 "Methods in Computational Linguistics" and 2. one module to be chosen from the catalogue MCL 6 of the program of study M.Sc. Computational Linguistics.		
14. Literatur:	Daniel Jurafsky and James H. Martin, 2008, Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 464601 Deutschsprachige Lehrveranstaltungen Wintersemester • 464602 Deutschsprachige Lehrveranstaltungen Sommersemester • 464603 courses in english - winter semester • 464604 courses in english - summer semester 		

-
16. Abschätzung Arbeitsaufwand: Präsenzzeit: 112h,
Selbststudium: 240h
-
17. Prüfungsnummer/n und -name:
 - 46461 Vertiefungslinie Sprachverarbeitung (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0
 - V Vorleistung (USL-V), Sonstiges
-
18. Grundlage für ... :
-
19. Medienform:
-
20. Angeboten von:
-

Modul: 29380 Vertiefungslinie Theoretische Informatik und Wissenschaftliches Rechnen

2. Modulkürzel:	050420555	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:	Apl. Prof. Ulrich Hertrampf		
9. Dozenten:	<ul style="list-style-type: none"> • Volker Diekert • Stefan Funke • Ulrich Hertrampf • Miriam Mehl • Dirk Pflüger • Stefan Zimmer 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 4. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 4. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 2. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>Die Studierenden kennen über das Wissen der Grundlagenvorlesungen hinaus detaillierte Methoden zur Lösung von Problemstellungen in zwei Teilgebieten der Theoretischen Informatik oder des Wissenschaftlichen Rechnens. Sie sind in der Lage, die Eignung von Methoden für eine gegebene Fragestellung zu beurteilen, die gelernten Verfahren geeignet anzuwenden und Vorschläge zur Modifikation von Verfahren zu machen, um neue Problemklassen zu bearbeiten.</p>		
13. Inhalt:	<p>Es werden Vorlesungen mit Übungen im Umfang von 8 SWS besucht. Zum Vertiefungsmodul Theoretische Informatik und Wissenschaftliches Rechnen gehören u.a. folgende Veranstaltungen mit jeweils 4 SWS (Vorlesung mit Übung):</p> <ol style="list-style-type: none"> 1) Algorithmische Geometrie 2) Ausgewählte Kapitel des Wissenschaftlichen Rechnens 3) Automaten über unendlichen Objekten 4) Discrete Optimization (auf Englisch) 5) High Performance Computing (auf Englisch) 6) Konkrete Mathematik 7) Parallele Numerik 8) Quantencomputing 		
14. Literatur:	<p>Die empfohlene Literatur wird bei den jeweiligen Lehrveranstaltungen angegeben.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 293801 Deutschsprachige Lehrveranstaltungen Wintersemester • 293802 Deutschsprachige Lehrveranstaltungen Sommersemester • 293803 courses in english - winter semester • 293804 courses in english - summer semester 		

16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 84 Stunden Nachbearbeitungszeit: 276 Stunden Gesamtzeit: 360 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 29381 Vertiefungslinie Theoretische Informatik und Wissenschaftliches Rechnen (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0• V Vorleistung (USL-V), mündliche Prüfung, 45 Min.
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Institut für Formale Methoden der Informatik

Modul: 29390 Vertiefungslinie Verteilte Systeme

2. Modulkürzel:	051200555	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> • Kurt Rothermel • Frank Dürr • Muhammad Tariq 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:	Grundlegende Kenntnisse im Bereich der Systemkonzepte und - Programmierung (z.B. Kursnummer 021506) und der Rechnernetze (z.B. Kursnummer 021531).		
12. Lernziele:	<p>Die Studierenden haben vertiefende Kenntnisse im Bereich der Verteilten Systeme, der Mobilten Systeme sowie der Rechnernetze erworben. Sie können mit den erlernten Methoden wissenschaftliche Arbeiten in diesen Bereichen verstehen und sind somit in der Lage, eine Masterarbeit in diesen Themengebieten anzufertigen. Außerdem haben sie das Rüstzeug, die in diesen Bereichen existierende Technologien zu analysieren, zu bewerten und weiterzuentwickeln. Durch die Teilnahme an Praktika und Programmierübungen werden sie vorbereitet, das fundierte Methodenwissen in der Praxis anzuwenden.</p>		
13. Inhalt:	<p>Es werden Vorlesungen bzw. Vorlesungen mit Übungen zu den Themengebieten Verteilte Systeme, Mobile Systeme sowie Rechnernetze im Umfang von 8SWS besucht, die im die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis tragen und dort inhaltlich beschrieben werden. In dem Vertiefungsmodul Verteilte Systeme gehören hierzu die Veranstaltungen:</p> <ol style="list-style-type: none"> 1) Verteilte Systeme (Vorlesung mit Übung, 6 LP) 2) Mobile Computing (Vorlesung mit Übung, 6 LP) 3) Verteilte Algorithmen (Vorlesung, 3 LP) 4) Asynchronous Middleware Systems (Vorlesung, 3 LP) 5) Höhere Kommunikationskonzepte und -protokolle (Vorlesung, 3 LP) 6) Konzepte der Peer-to-Peer-Systeme (Vorlesung, 3 LP) 		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 293901 Deutschsprachige Lehrveranstaltungen Wintersemester • 293902 Deutschsprachige Lehrveranstaltungen Sommersemester • 293903 courses in english - winter semester • 293904 courses in english - summer semester 		
16. Abschätzung Arbeitsaufwand:			

17. Prüfungsnummer/n und -name: 29391 Vertiefungslinie Verteilte Systeme (PL), mündliche Prüfung,
45 Min., Gewichtung: 1.0

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Verteilte Systeme

Modul: 29400 Vertiefungslinie Visualisierung und Interaktive Systeme

2. Modulkürzel:	051900555	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Thomas Ertl		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Martin Fuchs • Niels Henze • Albrecht Schmidt • Daniel Weiskopf • Steffen Koch 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Vertiefungsmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> • Grundlegende Kenntnisse in Computergraphik und Bildverarbeitung (z.B. Computergraphik 10060 und Imaging Science 10170) 		
12. Lernziele:	<p>Die Studierenden haben vertiefte Kenntnisse in mehreren Bereichen der Visualisierung, Computergraphik und der interaktiven Systeme erworben und können mit den erlernten Methoden wissenschaftliche Arbeiten in diesem Bereich verstehen. Sie haben das notwendige Rüstzeug, um eine Masterarbeit im Gebiet der Visualisierung, Computergraphik und der interaktiven Systeme anzufertigen.</p>		
13. Inhalt:	<p>Es werden Vorlesungen bzw. Vorlesungen mit Übungen zur Visualisierung und zu Interaktiven Systemen im Umfang von 8 SWS besucht, die im MINF-Katalog (MINF 1-8) den entsprechenden Verwendungshinweis tragen und dort inhaltlich beschrieben werden. In dem Vertiefungsmodul Visualisierung und Interaktive Systeme gehören hierzu die Veranstaltungen:</p> <ul style="list-style-type: none"> • Visual Computing (Lecture and Exercise in winter semester, 4SWS) • Information Visualization (Lecture and Exercise in winter semester, 4SWS) • Multimodal Interaction for Ubiquitous Computers (Lecture and Exercise in winter semester, 4SWS) • Image Synthesis / Bildsynthese (Lecture and Exercise in summer semester, 4SWS) • Scientific Visualization (Lecture and Exercise in summer semester, 4SWS) • Geometric Modeling and Animation / Geometrische Modellierung und Animation (Lecture and Exercise in summer semester, 4SWS) 		
14. Literatur:	Andrew S. Glassner, Principles of Digital Image Synthesis, 1995		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 294001 Deutschsprachige Lehrveranstaltungen Wintersemester • 294002 Deutschsprachige Lehrveranstaltungen Sommersemester 		

- 294003 courses in english - winter semester
- 294004 courses in english - summer semester

16. Abschätzung Arbeitsaufwand: Präsenzzeit: 84 Stunden
Selbststudium: 276 Stunden

17. Prüfungsnummer/n und -name:

- 29401 Vertiefungslinie Visualisierung und Interaktive Systeme (PL), mündliche Prüfung, 45 Min., Gewichtung: 1.0
- V Vorleistung (USL-V), Studienbegleitend

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Visualisierung und Interaktive Systeme

Modul: 29500 Visual Computing

2. Modulkürzel:	051900014	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Martin Fuchs		
9. Dozenten:	Martin Fuchs		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2009, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Ergänzende Spezialisierungsmodule</p> <p>M.Sc. Informatik, PO 2012, 1. Semester → Spezialisierungsmodule -->MINF →</p>		
11. Empfohlene Voraussetzungen:	Modul 051900002 Computergraphik		
12. Lernziele:	<p>The students know theoretical foundations for visual computing and acquired practical expertise in its core techniques. They are able to acquire scenes with digital cameras, can model their behavior and create content for non-2D displays and camera-projector systems.</p>		
13. Inhalt:	<p>The class is concerned with the digital processing of visual information by means of computer vision, computer graphics and image processing. It covers the following three interlocking topic complexes:</p> <p>Image processing:</p> <ul style="list-style-type: none"> • mathematical basics of image representations • noise models and noise suppression (including morphological, bilateral, and non-local filters) • selected topics from discrete image processing on image regions (e.g. photo montage with graph cuts, texture synthesis and space-time video completion) <p>Measuring / displaying light:</p> <ul style="list-style-type: none"> • selected topics from simple optics (esp. thin lenses and their interactions with light) • geometric camera models and calibration, typical optical distortions and means to counter them • radiometric camera calibration and HDR imaging • measuring and displaying color • plenoptic imaging / integral photography techniques, light field rendering and light field displays • passive stereo <p>Combined camera / illumination systems</p> <ul style="list-style-type: none"> • camera - illumination systems and photometric stereo • active stereo and projector-camera systems • the light transport matrix, its measurement and applications <p>Throughout, the class equally covers both acquisition (camera) and displays systems.</p>		

14. Literatur:	<ul style="list-style-type: none">• Andrew S. Glassner, Principles of Digital Image Synthesis, 1995• J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990• Jähne, Bernd, Digitale Bildverarbeitung, 2005• Literatur, siehe Webseite zur Veranstaltung• M. Pharr, G. Humphreys, Physically Based Rendering, 2004
15. Lehrveranstaltungen und -formen:	295001 Vorlesung mit Übungen Visual Computing
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden Gesamt: 180 Stunden
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 29501 Visual Computing (PL), schriftlich oder mündlich, 120 Min., Gewichtung: 1.0, Schriftliche Prüfung von 120 Min. oder mündlichen 30 Min• V Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	

Modul: 34940 Weiterführende Numerik partieller Differentialgleichungen

2. Modulkürzel:	080803802	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	unregelmäßig
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Kunibert Gregor Siebert		
9. Dozenten:	<ul style="list-style-type: none"> • Bernard Haasdonk • Christian Rohde • Kunibert Gregor Siebert 		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Informatik, PO 2012, 5. Semester → Ergänzungsmodule -->Wahlmodule aus Master Informatik →</p> <p>B.Sc. Informatik, PO 2012, 5. Semester → Vorgezogene Master-Module</p> <p>M.Sc. Informatik, PO 2012, 3. Semester → Ergänzende Spezialisierungsmodule</p>		
11. Empfohlene Voraussetzungen:	empfohlen: Einführung in die Numerik partieller Differentialgleichungen		
12. Lernziele:	Die Studenten verfügen über Kenntnis weiterführender Konzepte, Algorithmen und Methoden zur Lösung von partiellen Differentialgleichungen; sie erwerben die Fähigkeit, mit den erlernten Kenntnissen selbständig Methoden zu entwickeln, zu analysieren und umzusetzen, mit denen anwendungsorientierte Probleme effizient und genau gelöst werden		
13. Inhalt:	Vertiefende Themen der Numerik für PDEs, beispielsweise aus dem Bereich der Spektralmethoden, Finite Volumen, Continuous und Discontinuous Galerkin, schnelle Löser für dünnbesetzte Systeme, Mehrgitter und Multilevelverfahren, Anwendungen in der Kontinuumsmechanik, hierarchische Ansätze		
14. Literatur:	<p>D. Braess, Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie.</p> <p>D. Kröner, Numerical Schemes for Conservation Laws.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 349401 Vorlesung Weiterführende Numerik partieller Differentialgleichungen • 349402 Übung Weiterführende Numerik partieller Differentialgleichungen 		
16. Abschätzung Arbeitsaufwand:	<p>Insgesamt 270 h, wie folgt: Präsenzzeit: 42 h (V), 21 h (Ü) Selbststudium: 207</p>		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 34941 Weiterführende Numerik partieller Differentialgleichungen (PL), mündliche Prüfung, 30 Min., Gewichtung: 1.0 • V Vorleistung (USL-V), schriftlich, eventuell mündlich 		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

400 Schlüsselqualifikationen fachaffin

Zugeordnete Module: 10290 Projekt-INF
 38610 Seminar-INF 1
 42400 Seminar-INF 2

Modul: 10290 Projekt-INF

2. Modulkürzel:	051900095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes Semester
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Martin Fuchs		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Informatik, PO 2009, 5. Semester → Schlüsselqualifikationen fachaffin B.Sc. Informatik, PO 2012, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	Basismodule der Informatik. Darüber hinaus variabel je nach Projektanforderung.		
12. Lernziele:	<p>Die Studierenden sollen frühzeitig und beispielhaft an Informatik-Forschung herangeführt werden („undergraduate research“). Dazu soll in einem Team von mindestens 3 Studierenden in einem Zeitraum von höchstens 6 Monaten ein Projekt bearbeitet werden, das sich an aktuellen Forschungsfragestellungen der Abteilungen und Institute orientiert. Ein Beitrag zu laufenden Drittmittelprojekten ist möglich, ebenso eine Fortsetzung des Projekts in ausgewählten Bachelor-Arbeiten. Die Teilnehmer können ein forschungsorientiertes Projekt unter Anleitung planen, durchführen und die Ergebnisse dokumentieren und präsentieren.</p> <p>Teilnehmer verfügen insbesondere über die folgenden generischen Kompetenzen: Sie können in Teams an einem gemeinsamen Vorhaben arbeiten und ihre Beiträge den übergeordneten Erfordernissen anpassen. Sie sind in der Lage, ihre Ergebnisse den Projektteilnehmern vorzustellen und zu diskutieren und sie dabei gegebenenfalls auch fachfremden Teilnehmern zu erläutern. Sie können moderne Präsentations- und Visualisierungstechniken erfolgreich einsetzen.</p>		
13. Inhalt:	<p>Variabel: Es werden Projekte zu aktuellen Forschungsfragestellungen von den Prüfern des Fachbereichs Informatik angeboten. Die Themen haben einen überwiegenden Forschungscharakter, was sich aus dem Publikationspotential der erwarteten Ergebnisse ergibt. Die Projekte umfassen in der Regel: Einarbeitung und Literatursuche, Methodenentwicklung, Implementierung, Analyse, Dokumentation und Präsentation der Ergebnisse.</p> <p>Um dem Forschungscharakter des Projekts gerecht zu werden, soll das Ergebnis in einer wissenschaftlichen Publikation (max. 10 Seiten in Englisch) festgehalten werden, die einer einheitlichen Formatvorlage folgt. Einmal pro Semester sollen die bis zu einem Stichtag abgegebenen Projektpapiere auf einer internen Konferenz in einem Kurzbeitrag von den Studierenden präsentiert werden.</p>		
14. Literatur:	wird in der Veranstaltung und im Web bekannt gegeben.		
15. Lehrveranstaltungen und -formen:	102901 Seminar Projekt		
16. Abschätzung Arbeitsaufwand:	180 Stunden pro Teammitglied		

17. Prüfungsnummer/n und -name: 10291 Projekt-INF (USL), Sonstiges, Gewichtung: 1.0, Scheinkriterien: Aktive Mitwirkung im Projektteam. Abgabe eines Projektberichts in Form einer wissenschaftlichen Publikation (max. 10 Seiten in Englisch) gemäß einer einheitlichen Formatvorlage. Teilnahme und Mitwirkung an der internen Semesterkonferenz.

18. Grundlage für ... :

19. Medienform:

20. Angeboten von: Institut für Visualisierung und Interaktive Systeme

Modul: 38610 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:	B.Sc. Informatik, PO 2012, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	Basismodule der Informatik, darüber hinaus variabel: Je nach dem gewählten Seminarthema können Vorkenntnisse aus weiteren Vorlesungen benötigt werden.		
12. Lernziele:	Die Studierenden können sich mit wissenschaftlicher Originalliteratur auseinandersetzen, deren Kernaussagen rezipieren und sich ein spezielles Thema überwiegend im Selbststudium erarbeiten. Sie sind fähig relevante Daten zu sammeln und zu interpretieren und ihre Erkenntnisse einem Fach- und Laienpublikum verständlich zu präsentieren und auf Fragen aus dem Publikum angemessen und sachgerecht zu reagieren. Sie haben gelernt, sich mit einem wissenschaftlichen Thema über einen längeren Zeitraum hinweg auseinander zu setzen und eigenständig aktuelle Hintergrundinformation zu beschaffen. Sie haben generische Kompetenzen erworben, etwa aktiv an einer wissenschaftlichen Diskussion zu einem vorher bekannten Thema teilzunehmen und durch Fragen an den Vortragenden ihr Verständnis zu erweitern. Sie können eine Diskussion leiten und moderieren und sind befähigt, ihre Ergebnisse den Seminarteilnehmern vorzustellen und mit Hilfe moderner Präsentationstechniken zu visualisieren.		
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:	386101 Seminar		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17. Prüfungsnummer/n und -name:	38611 Seminar-INF 1 (BSL), Sonstiges, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			

Modul: 42400 Seminar-INF 2

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:	B.Sc. Informatik, PO 2012, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	Basismodule der Informatik, darüber hinaus variabel: Je nach dem gewählten Seminarthema können Vorkenntnisse aus weiteren Vorlesungen benötigt werden.		
12. Lernziele:	Die Studierenden können sich mit wissenschaftlicher Originalliteratur auseinandersetzen, deren Kernaussagen rezipieren und sich ein spezielles Thema überwiegend im Selbststudium erarbeiten. Sie sind fähig relevante Daten zu sammeln und zu interpretieren und ihre Erkenntnisse einem Fach- und Laienpublikum verständlich zu präsentieren und auf Fragen aus dem Publikum angemessen und sachgerecht zu reagieren. Sie haben gelernt, sich mit einem wissenschaftlichen Thema über einen längeren Zeitraum hinweg auseinander zu setzen und eigenständig aktuelle Hintergrundinformation zu beschaffen. Sie haben generische Kompetenzen erworben, etwa aktiv an einer wissenschaftlichen Diskussion zu einem vorher bekannten Thema teilzunehmen und durch Fragen an den Vortragenden ihr Verständnis zu erweitern. Sie können eine Diskussion leiten und moderieren und sind befähigt, ihre Ergebnisse den Seminarteilnehmern vorzustellen und mit Hilfe moderner Präsentationstechniken zu visualisieren.		
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:	424001 Seminar		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17. Prüfungsnummer/n und -name:	42401 Seminar-INF 2 (BSL), Sonstiges, Gewichtung: 1.0		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Institut für Softwaretechnologie		

Modul: 81100 Bachelorarbeit Informatik

2. Modulkürzel:	050525002	5. Moduldauer:	1 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes Semester
4. SWS:	8.0	7. Sprache:	Nach Ankuendigung
8. Modulverantwortlicher:		Univ.-Prof. Stefan Funke	
9. Dozenten:			
10. Zuordnung zum Curriculum in diesem Studiengang:		B.Sc. Informatik, PO 2009, 6. Semester B.Sc. Informatik, PO 2012, 6. Semester	
11. Empfohlene Voraussetzungen:			
12. Lernziele:			
13. Inhalt:			
14. Literatur:			
15. Lehrveranstaltungen und -formen:			
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:			
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:			