



Universität Stuttgart

Modulhandbuch
Studiengang Bachelor of Science Informatik
Prüfungsordnung: 2009

Universität Stuttgart
Keplerstr. 7
70174 Stuttgart

Inhaltsverzeichnis

Präambel	4
100 Basismodule	5
12060 Datenstrukturen und Algorithmen	6
10190 Mathematik für Informatiker und Softwaretechniker	8
10260 Programmierkurs	10
10280 Programmierung und Software-Entwicklung	12
10930 Technische Grundlagen der Informatik	14
10940 Theoretische Grundlagen der Informatik	16
200 Kernmodule	18
10020 Algorithmik	19
14910 Berechenbarkeit und Komplexität	21
17210 Einführung in die Softwaretechnik	23
10210 Mensch-Computer-Interaktion	25
10220 Modellierung	27
10240 Numerische und Stochastische Grundlagen	29
10270 Programmierparadigmen	31
10310 Rechnerorganisation	33
10330 Systemkonzepte und -programmierung	35
300 Ergänzungsmodule	37
320 Katalog ISG 1-3	38
10030 Architektur von Anwendungssystemen	39
10080 Datenbanken und Informationssysteme	41
10090 Grundlagen der Eingebetteten Systeme	43
10100 Grundlagen der Graphischen Ingenieursysteme	44
10110 Grundlagen der Künstlichen Intelligenz	46
10120 Grundlagen der Modellbildung und Simulation	48
10140 Grundlagen der Rechnerarchitektur	50
10150 Grundlagen des Compilerbaus und der Programmiersprachen	52
10170 Imaging Science	54
10250 Parallele Systeme	56
11110 Verteilte Systeme	57
330 Katalog ISW 1-3	59
10050 Bildverstehen	60
10060 Computergraphik	62
11900 Design and Test of Systems on a Chip	64
10170 Imaging Science	66
10180 Information Retrieval und Text Mining	68
10250 Parallele Systeme	69
10300 Rechnernetze	70
340 Katalog ISW 4-7	72
10160 Hardwarebeschreibungssprachen	73
350 Wahlmodule aus Master Informatik	75
10040 Bildsynthese	76
29570 Computer Interface Technologien	78
29580 Datenkompression	79
29590 Digitaler Systementwurf I	80
29440 Geometrische Modellierung und Animation	81
14380 Hardware Verification and Quality Assessment	83
29640 Mikrocontroller	85

29500 Visual Computing	87
11330 Visualisierung	89
400 Schlüsselqualifikationen fachaffin	91
10290 Projekt-INF	92
10320 Seminar-INF	94

Präambel

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

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

100 Basismodule

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

Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Erhard Plödereder		
9. Dozenten:	Stefan Funke		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 2. Semester → Basismodule BA (Komb) Informatik, 2. Semester → Module im Nebenfach		
11. 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) • 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 207 Stunden
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein.
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	12061 Datenstrukturen und Algorithmen
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Mathematik, 2. Semester → Nebenfach → Nebenfach Informatik B.Sc. Softwaretechnik, 2. Semester → Basismodule B.Sc. Wirtschaftsinformatik, 2. Semester → Informatik (B 1) B.Sc. Mechatronik, 2. Semester → Kernmodule B.Sc. Maschinelle Sprachverarbeitung, 2. Semester → Basismodule B.Sc. Technikpädagogik, 2. Semester → Hauptfach Informatik → Basismodule Informatik B.Sc. Technikpädagogik, 2. Semester → Wahlpflichtfach → Informatik M.Sc. Technikpädagogik, 2. Semester → Wahlpflichtfach B → Affines Wahlpflichtfach Informatik → Affines Wahlpflichtfach Informatik Basismodule

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:	Wolfgang Rump		
9. Dozenten:	Wolfgang Rump		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule		
11. 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 126 Stunden Nachbearbeitungszeit: 414 Stunden		
17a. Studienleistung:	Ein Übungsschein aus den beiden Veranstaltungen, jeweils im 1. oder 2. Fachsemester zu erwerben.		
17b. Prüfungsleistungen:	Schriftliche Prüfung: Zweistündige Klausur.		
18. Grundlage für ... :			
19. Medienform:			

20. Prüfungsnummer/n und -name: 10191 Mathematik für Informatiker und Softwaretechniker

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 1. Semester
 - Basismodule
- B.Sc. Technikpädagogik, 1. Semester
 - Hauptfach Informatik
 - Basismodule Informatik

Modul: 10260 Programmierkurs

2. Modulkürzel:	051520010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Jochen Ludewig		
9. Dozenten:	<ul style="list-style-type: none"> • • Sascha Riexinger • Holger Röder 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule BA (Komb) Informatik, 1. Semester → Module im Nebenfach		
11. Voraussetzungen:	Keine		
12. Lernziele:	Selbstständiges Erstellen von Programmen und Lösung von Programmieraufgaben in einer vorgegebenen Programmiersprache wie Java.		
13. Inhalt:	<p>Der Programmierkurs ergänzt die Vorlesung Programmierung und Software-Entwicklung (PSE). Die Teilnehmer erlernen eine weitere Programmiersprache (Java). Ihre Merkmale, Syntax und Semantik, werden denen der in PSE gelehrt Sprache gegenübergestellt. Praktische Übungen bereiten die Teilnehmer auf die Bearbeitung der Schein-Aufgabe vor.</p> <p>Die Lehrveranstaltung findet in zwei Varianten statt. Die Teilnahme richtet sich nach dem Studiengang:</p> <p>S. Riexinger:</p> <ul style="list-style-type: none"> • BSc. Informatik • BA (Komb) Informatik • BSc. Maschinelle Sprachverarbeitung <p>H. Röder:</p> <ul style="list-style-type: none"> • BSc. Softwaretechnik • BSc. Wirtschaftsinformatik • BSc. Technikpädagogik • MSc. Technikpädagogik 		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	102601 Übung Programmierkurs		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17a. Studienleistung:	USL (Übungsschein - Scheinkriterien werden zu Beginn der Veranstaltung angekündigt.)		
17b. Prüfungsleistungen:	-		
18. Grundlage für ... :			

19. Medienform:

20. Prüfungsnummer/n und -name: 10261 Programmierkurs

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 1. Semester
 - Basismodule
- B.Sc. Wirtschaftsinformatik, 1. Semester
 - Informatik (B 1)
- B.Sc. Maschinelle Sprachverarbeitung, 1. Semester
 - Basismodule
- B.Sc. Technikpädagogik, 1. Semester
 - Hauptfach Informatik
 - Basismodule Informatik
- B.Sc. Technikpädagogik, 1. Semester
 - Wahlpflichtfach
 - Informatik
- M.Sc. Technikpädagogik, 1. Semester
 - Wahlpflichtfach B
 - Affines Wahlpflichtfach Informatik
 - Affines Wahlpflichtfach Informatik Basismodule

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:	Bernhard Mitschang		
9. Dozenten:	Bernhard Mitschang		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule BA (Komb) Informatik, 1. Semester → Module im Nebenfach		
11. Voraussetzungen:	Keine. Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Teilnehmer haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> • Algorithmen und funktionale Programmierung Kap. 1 verwendet nur die funktionale Teilmenge der Programmiersprache Ada, keine Variablen, keine Prozeduren. Grammatik, Formale Sprachen und BNF werden eingeführt. • Imperative Programmierung Kap. 2 erweitert die verwendete Sprache durch die prozeduralen Konzepte, also Variablen und Prozeduren. Zu den Sprachkonstrukten werden Vor- und Nachbedingungen, mit den Schleifen die Invarianten eingeführt. Datentypen werden schrittweise ausgebaut. In Zusammenhang mit den Zeigern werden die Konzepte für Keller und Halde vermittelt. Die Entwicklung einfacher Programme wird gezeigt und geübt. • Aufbau und Organisation komplexer Programme. Die Modularisierung, die bei größeren Programmen notwendig ist, führt zur Kapselung und zu den abstrakten Datentypen. Damit entsteht die Möglichkeit, neue Datenstrukturen und Datentypen sicher zu definieren. Die Konzepte der Kompilation und der Interpretation werden erläutert. Wichtige Beispiele komplexer Datentypen werden entwickelt. Die Konzepte der Generalisierung (generische Einheiten) werden vermittelt. • Ausnahmebehandlung Möglichkeiten und Probleme der Ausnahmebehandlung sind Gegenstände dieses kurzen Kapitels. • Objektorientierte Programmierung Am Ende des Semesters steht ein Ausblick in die objektorientierte Programmierung, d.h. die Umsetzung der bereits bekannten Konzepte (ADTs) in die objektorientierte Sichtweise und die Vererbung. Dieser Teil bereitet die Programmierung in einer objektorientierten Sprache (3. Semester) vor. 		
14. Literatur:	Manuskripte: V.Claus (WS 08/09 bis SS 2009)		

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

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

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

15. Lehrveranstaltungen und -formen: • 102801 Vorlesung Programmierung und Softwareentwicklung
 • 102802 Übung Programmierung und Softwareentwicklung

16. Abschätzung Arbeitsaufwand: Präsenzzeit: 63 Stunden
 Vor-/Nachbearbeitungszeit: 187 Stunden
 Prüfungsvorbereitung: 20 Stunden

17a. Studienleistung: Studienleistung: Übungsschein, Vor. 3 mal vortragen in den
 Übungen und mindestens 50% der Übungspunkte erwerben,
 Teilnahme an den Zwischenklausuren.

17b. Prüfungsleistungen: Modulprüfung: Schriftliche Prüfung von 120 Minuten Dauer

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10281 Programmierung und Software-Entwicklung

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 1. Semester
 → Basismodule
- B.Sc. Wirtschaftsinformatik, 1. Semester
 → Informatik (B 1)
- B.Sc. Mechatronik, 1. Semester
 → Kernmodule
- B.Sc. Maschinelle Sprachverarbeitung, 1. Semester
 → Basismodule
- B.Sc. Technikpädagogik, 1. Semester
 → Hauptfach Informatik
 → Basismodule Informatik
- B.Sc. Technikpädagogik, 1. Semester
 → Wahlpflichtfach
 → Informatik
- M.Sc. Technikpädagogik, 1. Semester
 → Wahlpflichtfach B
 → Affines Wahlpflichtfach Informatik
 → Affines Wahlpflichtfach Informatik Basismodule
- B.Sc. Simulation Technology, 1. Semester
 → Grundstudium
- ohne Absch Lehramt-Pool, 1. Semester
 → Pflichtmodule

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:	Martin Radetzki		
9. Dozenten:	<ul style="list-style-type: none"> • Martin Radetzki • Sven Simon 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule		
11. 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:	<p>Elektrotechnische Grundlagen:</p> <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. <p>Digitaltechnische Komponenten:</p> <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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden Nachbearbeitungszeit: 117 Stunden		

17a. Studienleistung:	Prüfungsvorleistung: Teilnahme an einer Mindestzahl der Übungen, die zu Beginn der Lehrveranstaltung festgelegt wird.
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	10931 Technische Grundlagen der Informatik
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Technikpädagogik, 1. Semester → Hauptfach Informatik → Kernmodule Informatik M.Sc. Technikpädagogik, 1. Semester → Wahlpflichtfach B → Affines Wahlpflichtfach Informatik → Affines Wahlpflichtfach Informatik Kernmodule

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:	Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> • Ulrich Hertrampf • Volker Diekert 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule		
11. Voraussetzungen:	Keine		
12. Lernziele:	<ul style="list-style-type: none"> • Logik und Diskrete Strukturen: Die Studierenden haben die grundsätzlichen Kenntnisse in Logik und Diskreter Mathematik erworben, wie sie in den weiteren Grundvorlesungen der Informatik in verschiedenen Bereichen benötigt werden. • Automaten und Formale Sprachen: Die Studierenden beherrschen wichtige theoretische Grundlagen der Informatik, insbesondere die Theorie und Algorithmik endlicher Automaten. Hierzu gehört das Kennenlernen, Einordnung und Trennung der Chomskyschen Sprachklassen. 		
13. Inhalt:	<ul style="list-style-type: none"> • Logik und Diskrete Strukturen: Einführung in die Aussagenlogik; formale Sprache; Semantik (Wahrheitswerte); Syntax (Axiome und Schlussregeln); Normalformen; Hornformeln; aussagenlogische Resolution; Korrektheit und Vollständigkeit für die Aussagenlogik; Einführung in die Prädikatenlogik 1. Stufe; formale Sprache; Semantik und Syntax; Normalformen; Herbrand-Theorie; prädikatenlogische Resolution; Kombinatorik, Graphen, elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, RSA-Verfahren. • Automaten und Formale Sprachen: Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 84 Stunden		

Nachbearbeitungszeit: 276 Stunden

17a. Studienleistung: Prüfungsvorleistung: Übungsschein.

17b. Prüfungsleistungen: Schriftliche Prüfung von 120 Minuten Dauer.

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10941 Theoretische Grundlagen der Informatik

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 1. Semester
 - Basismodule
- B.Sc. Maschinelle Sprachverarbeitung, 1. Semester
 - Basismodule
- B.Sc. Technikpädagogik, 1. Semester
 - Hauptfach Informatik
 - Kernmodule Informatik
- B.Sc. Technikpädagogik, 1. Semester
 - Wahlpflichtfach
 - Informatik
- M.Sc. Technikpädagogik, 1. Semester
 - Wahlpflichtfach B
 - Affines Wahlpflichtfach Informatik
 - Affines Wahlpflichtfach Informatik Kernmodule
- ohne Absch Lehramt-Pool, 1. Semester
 - Pflichtmodule

200 Kernmodule

Zugeordnete Module:

- 10020 Algorithmmik
- 14910 Berechenbarkeit und Komplexität
- 17210 Einführung in die Softwaretechnik
- 10210 Mensch-Computer-Interaktion
- 10220 Modellierung
- 10240 Numerische und Stochastische Grundlagen
- 10270 Programmierparadigmen
- 10310 Rechnerorganisation
- 10330 Systemkonzepte und -programmierung

Modul: 10020 Algorithmik

2. Modulkürzel:	050420015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> • Ulrich Hertrampf • Volker Diekert 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 5. Semester → Kernmodule		
11. 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Übungsschein		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10021 Algorithmik		

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Wirtschaftsinformatik, 5. Semester
 → Wahlpflichtbereich (Bereich C)
 → Wahlbereich II: Informatik Basis

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:	Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> • Stefan Funke • Volker Diekert • Ulrich Hertrampf 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Kernmodule		
11. 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42h Nachbearbeitungszeit: 118h Prüfungsvorbereitung: 20h Gesamt: 180h		
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein.		
17b. Prüfungsleistungen:	Modulprüfung: Schriftliche Prüfung, 120 Minuten		
18. Grundlage für ... :	10020 Algorithmik		
19. Medienform:			
20. Prüfungsnummer/n und -name:	14911 Berechenbarkeit und Komplexität		
21. Angeboten von:			

22. Zuordnung zu weiteren Curricula:
- B.Sc. Mathematik, 5. Semester
 - Ergänzungsmodule
 - B.Sc. Simulation Technology, 3. Semester
 - Fachstudium
 - Vertiefungsrichtung CS
-

Modul: 17210 Einführung in die Softwaretechnik

2. Modulkürzel:	051520015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Jochen Ludewig		
9. Dozenten:	Jochen Ludewig		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Kernmodule		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051510005 Datenstrukturen und Algorithmen sowie entsprechende Programmiererfahrung		
12. Lernziele:	EST ist, wie der Name sagt, die allgemeine Einführung in die Softwaretechnik. Sie ist abgestimmt auf die Software-Qualität im 1. und Programmentwicklung im 3. Semester. Die Teilnehmer kennen die Grundbegriffe der Softwaretechnik und haben wichtige Techniken des Softwareprojekt-Managements und der Software-Entwicklung erlernt.		
13. Inhalt:	Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung, wie sie in der Praxis stattfindet. Die einzelnen Themen sind: <ul style="list-style-type: none"> • Abgrenzung und Motivation des Software Engineerings • Vorgehensmodelle; Software-Management; Software-Prüfung und Qualitätssicherung • Methoden, Sprachen und Werkzeuge für die einzelnen Phasen: Spezifikation, Grobentwurf, Feinentwurf, Codierung, Test		
14. Literatur:	Ludewig, Lichter: Software Engineering. dpunkt-Verlag, Heidelberg. 2. Aufl. 2010		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 172101 Vorlesung Einführung in die Softwaretechnik • 172102 Übung Einführung in die Softwaretechnik 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Übungsschein		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 60 Minuten Dauer		
18. Grundlage für ... :	<ul style="list-style-type: none"> • 16500 Software Engineering • 16510 Software-Praktikum 		
19. Medienform:			
20. Prüfungsnummer/n und -name:	17211 Einführung in die Softwaretechnik		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 2. Semester → Basismodule		

B.Sc. Mechatronik, 5. Semester
→ Kernmodule

B.Sc. Technikpädagogik, 4. Semester
→ Hauptfach Informatik
→ Pflichtmodule Informatik

M.Sc. Technikpädagogik, 4. Semester
→ Wahlpflichtfach B
→ Affines Wahlpflichtfach Informatik
→ Affines Wahlpflichtfach Informatik Pflichtmodule

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:	Thomas Ertl		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Kernmodule		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051200005 Systemkonzepte und -programmierung 		
12. Lernziele:	<p>Verständnis der Methoden und Konzepte der Mensch-Computer-Interaktion, insbesondere der graphisch-interaktiven Systeme:</p> <ul style="list-style-type: none"> • Architektur und technische Funktionsweise von GUI-Systemen • kognitive Grundlagen und Konsequenzen für die Software-Ergonomie • praktische Erfahrung mit der Erstellung von Benutzungsoberflächen mit verschiedenen Programmierschnittstellen 		
13. Inhalt:	<ul style="list-style-type: none"> • Einführung: Bezug zu anderen Gebieten, historische Entwicklung • Menschliche Aspekte: sensorische (insbesondere) visuelle Wahrnehmung, Motorik, Gedächtnis-, Aufmerksamkeits- und Problemlösungsmodelle • Computer-Aspekte: Ein/Ausgabegeräte, Display-Architekturen und Event-Verarbeitung, Multimedia-Grundlagen (Vektor-/Rastergraphik, Audio/Video, Farbsysteme), 2D-Graphik (Compositing, Rasterisierung, Linien, Polygone, Text, Bilder, APIs) • Interaktionskonzepte und -stile: Geräte- vs. Task-Ebene, Kommandozeile, Menüs, Formulare, Gestik, Spracheingabe, graphische Stile: Direkte Manipulation, WYSIWYG, Icons • Fenstersysteme und GUI Toolkits Basisaspekte (Fenstermanagement, Event-Zuordnung), Schichtenaufbau (X, WPF), Widgets/Componenten, Toolkit-Architektur (z.B. Qt, AWT/Swing, XML/Web-basiert), Verwendung von Standardkomponenten • Software Ergonomie, Entwurfsprinzipien: Normen, Regeln (Shneidermann), Style Guides, Modelle (MVC), Metaphern, Entwicklungswerkzeuge • Evaluation: Experten-Evaluation (Walkthrough, GOMS), Benutzer-Evaluation (Think-aloud, Interviews, Auswertung) • Spezielle Systeme: mobile Geräte, Virtual/Augmented Reality <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"> • Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004 • Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005 • Bernhard Preim, Entwicklung interaktiver Systeme, 1999 		
15. Lehrveranstaltungen und -formen:	• 102101 Vorlesung Mensch-Computer-Interaktion		

• 102102 Übung Mensch-Computer-Interaktion

16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein.
17b. Prüfungsleistungen:	Modulprüfung: Schriftliche Prüfung, 120 Minuten.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	10211 Mensch-Computer-Interaktion
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	<ul style="list-style-type: none">B.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog ISGB.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog ISWB.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog SWTB.Sc. Wirtschaftsinformatik, 6. Semester<ul style="list-style-type: none">→ Informatik (B 1)B.Sc. Wirtschaftsinformatik, 6. Semester<ul style="list-style-type: none">→ Wahlpflichtbereich (Bereich C)→ Wahlbereich II: Informatik BasisB.Sc. Wirtschaftsinformatik, 6. Semester<ul style="list-style-type: none">→ Wahlpflichtbereich (Bereich C)→ Wahlbereich III: Informatik SimulationB.Sc. Maschinelle Sprachverarbeitung, 5. Semester<ul style="list-style-type: none">→ Wahlbereich E/I

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:	Frank Leymann		
9. Dozenten:	<ul style="list-style-type: none"> • Bernhard Mitschang • Frank Leymann 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Kernmodule		
11. 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 • XML, DTD, XML-Schema, Info-Set, Namensräume, XSLT, XPath • Metamodelle & RepositoryMDA Konzepte • RDF, RDF-S & Ontologien • UML • Petri Netze, Workflownetze • BPMN • IT Landkarten (Modellierung komplexe Systeme - FMC) 		
14. Literatur:	<ul style="list-style-type: none"> • A. Knöpfel, B. Gröne, P. Tabeling, Fundamental Modeling Concepts - Effective Communication of IT Systems, 2005 • A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002 • B. Daum, U. Merten, System Architecture With XML, 2003 • M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work - Objektorientierte Modellierung mit UML2, 2005 • P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure, Semantic Web, 2008 • T.J. Teorey, Database Modeling & Design, 2nd Edition,, 1994 • V. Gruhn, D. Pieper, C. Röttgers, MDA, 2006 • W. van der Aalst, K. van Hee, Workflow Management, 2002 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102201 Vorlesung Modellierung • 102202 Übung Modellierung 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10221 Modellierung		

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:
- B.Sc. Softwaretechnik, 4. Semester
 - Ergänzungsmodule
 - Katalog ISG
 - B.Sc. Wirtschaftsinformatik, 4. Semester
 - Informatik (B 1)
 - B.Sc. Technikpädagogik, 4. Semester
 - Hauptfach Informatik
 - Pflichtmodule Informatik
 - M.Sc. Technikpädagogik, 4. Semester
 - Wahlpflichtfach B
 - Affines Wahlpflichtfach Informatik
 - Affines Wahlpflichtfach Informatik Pflichtmodule
-

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:	Marc Alexander Schweitzer		
9. Dozenten:	<ul style="list-style-type: none"> • Stefan Zimmer • Michael Bader • Marc Alexander Schweitzer 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Kernmodule		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 080300100 Mathematik für Informatiker und Softwaretechniker 		
12. Lernziele:	Beherrschung grundlegende Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.		
13. Inhalt:	<p>Methoden der angewandten Mathematik, insbesondere der Numerik, Stochastik und Statistik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse:</p> <ul style="list-style-type: none"> • numerischer Algorithmik • Gleitpunktzahlen und Gleitpunkarithmetik • Interpolation • 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. <p>Dabei wird ein konstruktiv-algorithmischer Zugang gewählt, der sich an konkreten Aufgabenstellungen aus der Informatik orientiert.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Peter Bastian, Numerische und Stochastische Grundlagen der Informatik, 2008 • Schickinger T., Steger A., Diskrete Strukturen, Band 2, 2002 • Schwarz, H.R., Köckler, N., Numerische Mathematik, 2004 		
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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 63 Stunden		

Nachbearbeitungszeit: 207 Stunden

17a. Studienleistung: Keine

17b. Prüfungsleistungen: Modulprüfung: Schriftliche Prüfung von 90 Minuten Dauer

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10241 Numerische und Stochastische Grundlagen

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Wirtschaftsinformatik, 5. Semester
→ Wahlpflichtbereich (Bereich C)
→ Wahlbereich III: Informatik Simulation

Modul: 10270 Programmierparadigmen

2. Modulkürzel:	051510010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Erhard Plödereder		
9. Dozenten:	<ul style="list-style-type: none"> • Viktor Avrutin • Erhard Plödereder 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Kernmodule		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Programmiererfahrung in einer ersten Programmiersprache • Modul 051520005 Programmierung und Softwareentwicklung • Modul 051520010 Programmierkurs • Modul 051510005 Datenstrukturen und Algorithmen 		
12. Lernziele:	Die Studierenden haben die Konzepte einer zweiten Programmiersprache ihrer Ausbildung verstanden und deren Grundbegriffe erlernt. Sie können ihre Kenntnisse in einfachen Programmen anwenden.		
13. Inhalt:	<p>Bezogen auf eine zweite Programmiersprache der Informatikausbildung behandelt die Lehrveranstaltung dieses Moduls die folgenden Inhalte:</p> <ul style="list-style-type: none"> • Grundbegriffe und Techniken der zweiten Programmiersprache, insbesondere Aufbau und Struktur der Programme sowie Abstraktions- und Ausführungskonzepte der Sprache • Umsetzung der objekt-orientierten und der generischen Programmierparadigmen • Umgang mit Standardsituationen der Programmierung • Auswirkungen und Anwendungsbereiche der behandelten Programmierparadigmen <p>Die Wahl der Programmiersprache wird vom Dozenten bekanntgegeben.</p>		
14. Literatur:	<p>Java ist auch eine Insel, Christian Ullendörfer, 1475 S., 8. Auflage, Galileo Computing 2009, ISBN 978-3-8362-1371-4, online-Version: http://openbook.galileocomputing.de/javainsel/</p> <p>Ken Arnold, James Gosling, David Holmes, The Java Programming Language, Fourth Edition, Addison-Wesley Professional, 2005, ISBN 0-321-34980-6</p>		
15. Lehrveranstaltungen und -formen:	102701 Übung Programmierparadigmen		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17a. Studienleistung:	Keine		
17b. Prüfungsleistungen:	studienbegleitende Abgabe von Programmierlösungen		
18. Grundlage für ... :			
19. Medienform:			

20. Prüfungsnummer/n und -name: 10271 Programmierparadigmen

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Simulation Technology, 3. Semester
→ Fachstudium
→ Vertiefungsrichtung CS

Modul: 10310 Rechnerorganisation

2. Modulkürzel:	051700005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	9.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Martin Radetzki • Sven Simon 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Kernmodule		
11. 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 Grundschaltungen 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"> • 103101 Vorlesung Rechnerorganisation 1 • 103102 Übung Rechnerorganisation 1 • 103103 Vorlesung Rechnerorganisation 2 • 103104 Hardwarepraktikum Rechnerorganisation 2
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 95 Stunden Nachbearbeitungszeit: 265 Stunden
17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Praktikum und Übungsaufgaben, 0.30 (Lehrveranstaltungsbegleitend) Schriftliche Prüfung von 90 Minuten Dauer, 0.70
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10311 Rechnerorganisation • 10312 Rechnerorganisation - Praktikum und Übungsaufgaben
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	

Modul: 10330 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:	Kurt Rothermel		
9. Dozenten:	<ul style="list-style-type: none"> • Kurt Rothermel • Frank Leymann 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Kernmodule		
11. 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"> • zentrale, verteilte, parallele Systeme • client/server, Producer/Consumer, P2P, Grid • Betriebssysteme, Systemplattformen (Middleware), Kerne für eingebettete Systeme <p>Grundlagen der Rechnernetze</p> <ul style="list-style-type: none"> • Grundlegende Netzkonzepte und -architekturen • Grundlegende Kommunikationsprotokolle und -dienste • Prozessmanagement • Prozessbeschreibung, -kontrolle • Threads <p>Interprozesskommunikation</p> <ul style="list-style-type: none"> • Gemeinsamer Speicher • Message Passing (Messages, RPC/RMI, Message Queuing, Ereignisse) <p>Synchronisation</p> <ul style="list-style-type: none"> • Abstraktionen für Shared Memory (Semaphore, Monitore, ...) • Abstraktionen für Message Passing • Verklemmungen (Modelle, Behandlung) <p>Kausalität und logische Uhren Scheduling</p> <ul style="list-style-type: none"> • Uniprozessor • Multiprozessor • Realtime 		

	Schutz und Sicherheit
	<ul style="list-style-type: none"> • grundlegende Konzepte der Sicherheit • ACLs, Capabilities
14. Literatur:	<ul style="list-style-type: none"> • Literatur, siehe Webseite zur Veranstaltung
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 103301 Vorlesung Systemkonzepte und -programmierung • 103302 Übung Systemkonzepte und -programmierung
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer (Faktor 0.7) Benoteter Übungsschein (Faktor 0.3)
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10331 Systemkonzepte und -programmierung • 10332 Systemkonzepte und -programmierung - Übungsschein
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	<p>B.Sc. Softwaretechnik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISG <p>B.Sc. Softwaretechnik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISW <p>B.Sc. Softwaretechnik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog SWT <p>B.Sc. Technikpädagogik, 3. Semester</p> <ul style="list-style-type: none"> → Hauptfach Informatik → Pflichtmodule Informatik <p>M.Sc. Technikpädagogik, 3. Semester</p> <ul style="list-style-type: none"> → Wahlpflichtfach B → Affines Wahlpflichtfach Informatik → Affines Wahlpflichtfach Informatik Pflichtmodule

300 Ergänzungsmodule

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

320 Katalog ISG 1-3

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10080	Datenbanken und Informationssysteme
	10090	Grundlagen der Eingebetteten Systeme
	10100	Grundlagen der Graphischen Ingenieursysteme
	10110	Grundlagen der Künstlichen Intelligenz
	10120	Grundlagen der Modellbildung und Simulation
	10140	Grundlagen der Rechnerarchitektur
	10150	Grundlagen des Compilerbaus und der Programmiersprachen
	10170	Imaging Science
	10250	Parallele Systeme
	11110	Verteilte Systeme

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:	Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 4. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	Am Ende des Moduls ist das Konzept einer Anwendungsarchitektur verstanden. Die Rolle von Middleware im Rahmen von Anwendungsarchitekturen ist klar. Grundsätzliche Strukturen und Muster in Anwendungsarchitekturen sind bekannt. Nicht-funktionale Eigenschaften von Anwendungssystemen und deren Bedeutung sind verstanden.		
13. Inhalt:	<ul style="list-style-type: none"> • "Anwendungsarchitektur": Terminologie, Rollen & Artefakte • Datenbanksysteme • Schichtenarchitekturen (Client-Server Cut, N-Tier) • API, RPC & Middleware • Transaktionen • Nachrichtenorientierung • Message-Oriented Middleware • TP Monitore (direct TP, queued TP) • QoS (high availability, scalability, security,...) • Komponenten & ihre Beschreibung (EJB & WSDL) • Application Server (JEE) • Architekturstile (POSA,...) • Lose Kopplung & SOA • WfMS: Programmieren im Grossen • Optional: Lizenzierungen, Portabilität, Deployment, Konfiguration, Installation 		
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 • M. Fowler, Patterns of Enterprise Application Architecture, 2003 • P. Bernstein, E. Newcomer, Principles of Transaction Processing, 1997 • S. Conrad, W. Hasselbring, A. Koschel, R. Tritsch, Enterprise Application Integration, 2006 • S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005 • W. Emmerich, Konstruktion von verteilten Objekten, 2003 		

15. Lehrveranstaltungen und -formen: • 100301 Vorlesung Grundlagen der Architektur von Anwendungssystemen
 • 100302 Übung Grundlagen der Architektur von Anwendungssystemen

16. Abschaetzung Arbeitsaufwand: Präsenzzeit: 42 Stunden
 Nachbearbeitungszeit: 138 Stunden

17a. Studienleistung: Prüfungsvorleistung: Übungsschein.

17b. Prüfungsleistungen: Schriftliche Prüfung von 90 Minuten Dauer.

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10031 Architektur von Anwendungssystemen

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Softwaretechnik, 6. Semester
 → Ergänzungsmodule
 → Katalog ISG

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:	Bernhard Mitschang		
9. Dozenten:	<ul style="list-style-type: none"> • Bernhard Mitschang • Holger Schwarz 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 3. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISG 1-3 <p>BA (Komb) Informatik, 5. Semester</p> <ul style="list-style-type: none"> → Module im Nebenfach → Katalog ISG 		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 052010001 Modellierung 		
12. Lernziele:	Die Studierenden haben die erforderlichen Kenntnisse für Datenbankprogrammierer in angemessenem Umfang erworben.		
13. Inhalt:	<p>Diese Vorlesung ist als Einstiegsvorlesung für das Fachgebiet Datenbanksysteme konzipiert. Es wird dabei versucht, das Fachgebiet so gründlich und umfassend darzustellen, wie es für den Datenbankprogrammierer erforderlich und angemessen erscheint.</p> <p>Stoffauswahl, -umfang und Detaillierungsgrad wurden deshalb aus der Sicht der Anwendung von Datenbanksystemen getroffen, wozu hauptsächlich das Verständnis von Datenmodellen, der Entwurf von logischen Datenbankstrukturen (DB-Schemata) und der Umgang mit Datenbanksprachen gehören. Weiterhin soll durch Stoffauswahl das Verständnis anderer Vorlesungen, die gewisse Querbezüge und Verbindungen zu Datenbanksystemen aufweisen, erleichtert werden. Dazu gehören zum Beispiel Vorlesungen über Informationssysteme, Systemanalyse, Wissensdarstellung, Expertensysteme, Multimedia-Datenbanksysteme oder Rechnergestützte Ingenieursysteme (CAD/CAM).</p> <p>Übersicht:</p> <ul style="list-style-type: none"> • Informationsmodelle (Entity-Relationship-Modell) • Datenmodelle (Relationenmodell, Netzwerk- und hierarchisches Datenmodell) • Relationale Anfragesprachen (SQL, Relationenalgebra und Relationenkalkül) • Logischer DB-Entwurf (Relationensynthese und Normalformen) • Netzwerk-Datenmodell und Hierarchisches Datenmodell <p>Stoffauswahl, -umfang und Detaillierungsgrad werden aus der Sicht der Anwendung von Datenbanksystemen getroffen.</p>		
14. Literatur:	<ul style="list-style-type: none"> • A. Kemper, A. Eickler, Datenbanksysteme - Eine Einführung, 2004 • C. Date, An Introduction to Database Systems, 2003 • H. Garcia-Molina, J. D. Ullman, J. Widom, Database Systems. The Complete Book, 2003 • R. Elmasri, S. Navathe, Fundamentals of Database Systems, 2003 		

15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none">• 100801 Vorlesung Grundlagen der Datenbanken und Informationssysteme• 100802 Übung Grundlagen der Datenbanken und Informationssysteme
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17a. Studienleistung:	Prüfungsvorleistung; Modalitäten werden in der ersten Vorlesung angegeben
17b. Prüfungsleistungen:	Schriftliche Prüfung von 60 Minuten Dauer.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	10081 Datenbanken und Informationssysteme
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 3. Semester → Ergänzungsmodule → Katalog ISG B.Sc. Maschinelle Sprachverarbeitung, 5. Semester → Wahlbereich E/I

Modul: 10090 Grundlagen der Eingebetteten Systeme

2. Modulkürzel:	051711010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Martin Radetzki		
9. Dozenten:	Martin Radetzki		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051700005 Rechnerorganisation 		
12. Lernziele:	Methodische und werkzeunterstützte Konstruktion von Hardware und Software für auf spezifische Anwendungen hin optimierte eingebettete Systeme.		
13. Inhalt:	Modelle zur Systemspezifikation, Modellierungssprachen und ihre Simulation, System- und Architektursynthese, Allokation von Ressourcen und Bindung von Aufgaben/Operationen, Hardware-Software-Partitionierung, Verfahren zur Ablaufplanung für parallele Architekturen, Optimierungsverfahren, Anwendungsspezifische Prozessoren, On-Chip/Board-Verbindungsnetzwerke.		
14. Literatur:	<ul style="list-style-type: none"> • 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"> • 100901 Vorlesung Grundlagen der Eingebetteten Systeme • 100902 Übung Grundlagen der Eingebetteten Systeme 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Keine.		
17b. Prüfungsleistungen:	Bewertete Rechnerübungen - vorlesungsbegleitend (0.25) Schriftliche Prüfung von 90 Minuten Dauer (0.75)		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> • 10091 Grundlagen der Eingebetteten Systeme • 10092 Grundlagen der Eingebetteten Systeme - Rechnerübungen 		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 4. Semester → Ergänzungsmodule → Katalog ISG B.Sc. Softwaretechnik, 4. Semester → Ergänzungsmodule → Katalog ISW		

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:	Dieter Roller		
9. Dozenten:	<ul style="list-style-type: none"> • • Dieter Roller 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 4. Semester → Module im Nebenfach → Katalog ISG		
11. 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 • 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	keine		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 60 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10101 Grundlagen der Graphischen Ingenieursysteme		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 6. Semester → Ergänzungsmodule		

- Katalog ISG
 - B.Sc. Softwaretechnik, 6. Semester
 - Ergänzungsmodule
 - Katalog ISW
-

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:	Gunther Heidemann		
9. Dozenten:	Gunther Heidemann		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. 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 • Lernen 		
14. Literatur:	<ul style="list-style-type: none"> • G. F. Luger, Künstliche Intelligenz, 2001 • S. Russell, P. Norvig, Künstliche Intelligenz, 2004 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101101 Vorlesung Grundlagen der Künstlichen Intelligenz • 101102 Übung Grundlagen der Künstlichen Intelligenz 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 60 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10111 Grundlagen der Künstlichen Intelligenz		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 5. Semester → Ergänzungsmodule → Katalog ISG		

- B.Sc. Softwaretechnik, 5. Semester
 - Ergänzungsmodule
 - Katalog ISW
 - B.Sc. Wirtschaftsinformatik, 5. Semester
 - Informatik (B 1)
 - B.Sc. Wirtschaftsinformatik, 5. Semester
 - Wahlpflichtbereich (Bereich C)
 - Wahlbereich II: Informatik Basis
 - B.Sc. Wirtschaftsinformatik, 5. Semester
 - Wahlpflichtbereich (Bereich C)
 - Wahlbereich III: Informatik Simulation
 - B.Sc. Maschinelle Sprachverarbeitung, 5. Semester
 - Wahlbereich E/I
-

Modul: 10120 Grundlagen der 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:	Marc Alexander Schweitzer		
9. Dozenten:	<ul style="list-style-type: none"> • Stefan Zimmer • Michael Bader • Marc Alexander Schweitzer 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 3. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISG 1-3 <p>BA (Komb) Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Module im Nebenfach → Katalog ISG 		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 080300100 Mathematik für Informatiker und Softwaretechniker • 051240005 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 kontinuierliche Modelle sowie deren numerische Behandlung. Vorgestellt werden insbesondere die Diskretisierung gewöhnlicher und partieller Differentialgleichungen mit Finiten Differenzen sowie die Lösung großer dünnbesetzter Gleichungssysteme, Eigenwert- und Minimierungsprobleme. Diese Verfahren werden auf Problemstellungen der Natur- und Ingenieurwissenschaften angewendet, z.B. Populationwachstum, Wärmetransport und Verkehrssimulation.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Modellbildung und Simulation - Eine anwendungsorientierte Einführung <p>Bungartz, H.-J., Zimmer, S., Buchholz, M., Pflüger, D., Springer Verlag, eXamen.press, 2009, ISBN 978-3-540-79809-5</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101201 Vorlesung Grundlagen der Modellbildung und Simulation • 101202 Übung Grundlagen der Modellbildung und Simulation 		
16. Abschaetzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden</p> <p>Nachbearbeitungszeit: 138 Stunden</p>		
17a. Studienleistung:	Keine		
17b. Prüfungsleistungen:	<p>Schriftliche Prüfung von 90 Minuten Dauer; bei geringer Teilnehmerzahl mündliche Prüfung von 30 Minuten Dauer.</p>		
18. Grundlage für ... :			

19. Medienform:

20. Prüfungsnummer/n und -name: 10121 Grundlagen der Modellbildung und Simulation

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 4. Semester
 - Ergänzungsmodule
 - Katalog ISG
- B.Sc. Softwaretechnik, 4. Semester
 - Ergänzungsmodule
 - Katalog ISW
- B.Sc. Softwaretechnik, 4. Semester
 - Ergänzungsmodule
 - Katalog SWT

Modul: 10140 Grundlagen der Rechnerarchitektur

2. Modulkürzel:	051700010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Hans-Joachim Wunderlich		
9. Dozenten:	Hans-Joachim Wunderlich		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051700005 Rechnerorganisation 		
12. Lernziele:	<ul style="list-style-type: none"> • Grundlegendes Verständnis der Herausforderungen bei Einsatz, Entwurf und Fertigung moderner Prozessoren • Kenntnis über den Zusammenhang von Kosten, Performanz und Verlustleistung beim Entwurf von Prozessoren und der Konzeption komplexer Systeme • Kenntnis und Bewertung aktueller Architekturkonzepte 		
13. Inhalt:	<p>Es werden die klassischen Themen der Rechnerarchitektur als Hardware/Software-Schnittstelle sowie weiterführende technologische Themen behandelt.</p> <p>Dazu gehören insbesondere:</p> <ul style="list-style-type: none"> • Technologische Grundlagen: Entwurstile und Fertigungstechnik, Ausbeute, Test und Zuverlässigkeit, Kosten und Qualität, Skalierung. • Performanz: Taktfrequenz und Befehle pro Takt, Geschwindigkeitsanalyse und -optimierung. • Verlustleistung: Verlustleistungsanalyse, Optimierung von Verlustleistung und Performanz, Verlustleistung und Skalierung. • Computerarithmetik: Effiziente Hardwareimplementierung der Grundrechenarten, Hardwareimplementierung spezieller Funktionen wie Exponentialfunktion, Logarithmus und trigonometrische Funktionen, Gleitkommaoperationen und Standards, arithmetische Pipelines und Filter, aktuelle Gleitkommaeinheiten wie SPE des Cell Prozessors oder SPARC. • Parallelität auf Befehlsebene: Superskalarrechner, statisches und dynamisches Scheduling, Out-of-Order Ausführung und VLIW-Rechner, Multithreading • Parallele Architekturen: Shared Memory und Message Passing, Multi-Core Prozessoren und Multi-Core Systeme auf einem Chip. • Speicherhierarchie: Speichertechnologie und Cacheentwurf. • Fehlertoleranztechniken: Einzelprozessoren und Mehrprozessorsysteme. 		
14. Literatur:	<ul style="list-style-type: none"> • I. Koren, Computer Arithmetic Algorithms, 2001 • J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 2006 • S. Iman, M. Pedram, Logic Synthesis for Low Power VLSI Designs, 1997 		

15. Lehrveranstaltungen und -formen: • 101401 Vorlesung Grundlagen der Rechnerarchitektur
 • 101402 Übung Grundlagen der Rechnerarchitektur

16. Abschaetzung Arbeitsaufwand: Präsenzzeit: 42 Stunden
 Nachbearbeitungszeit: 138 Stunden

17a. Studienleistung: Keine

17b. Prüfungsleistungen: Schriftliche Prüfung von 90 Minuten Dauer.

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10141 Grundlagen der Rechnerarchitektur

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Softwaretechnik, 5. Semester
 → Ergänzungsmodule
 → Katalog ISG

 B.Sc. Softwaretechnik, 5. Semester
 → Ergänzungsmodule
 → Katalog ISW

Modul: 10150 Grundlagen des Compilerbaus und der Programmiersprachen

2. Modulkürzel:	051510015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Erhard Plödereder		
9. Dozenten:	Erhard Plödereder		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	Grundkenntnisse aus den Einführungsvorlesungen des Informatikgrundstudiums, sowie einige Erfahrungen mit Programmierung. Vorkenntnisse über formale Sprachen sind vorteilhaft, aber nicht zwingend.		
12. Lernziele:	Die Studierenden haben die Grundkenntnisse erlangt, die zur effizienten Verwendung von Lexer- und Parser-Generatoren zur Analyse von Eingabetexten nötig sind. Sie haben gelernt, die Fehlermeldungen aus diesen Generatoren und den Compilern oder Interpretern richtig einzuordnen. Ferner haben sie durch Betrachtung der Implementierungsmodelle typischer Programmiersprachenkonstrukte Verständnis für das Ausführungsverhalten und für typische, gefährliche Fehlerquellen in Anwendungsprogrammen erlangt.		
13. Inhalt:	Compilerarchitekturen im Überblick; lexikalische und syntaktische Analyse von Texten mit formaler Grammatik, insb. von Programmiersprachen. Lexikalische Analyse: endliche Automaten und ihre Implementierung; Syntaxanalyse: diverse Parser-Strategien, ihre Implementierung und Eigenschaften. Methoden der automatischen Generierung von Analysatoren aus Spezifikationen der Grammatiken. Fehlererkennung und -behandlung. Analyse der statischen Semantik: Grundbegriffe und elementare Methoden. Eigenschaften von Programmiersprachen; Realisierung der Laufzeitsemantik prozeduraler Programmiersprachen aus Benutzersicht, insbesondere Implementierungsmodelle der Speicherverwaltung und der Unterprogrammaufrufe. Vermeidung typischer Fehlerquellen und überraschender Probleme in Anwendungsprogrammen.		
14. Literatur:	<ul style="list-style-type: none"> • Aho, Sethi, Ullman, Compilers - Principles, Techniques, and Tools, 1988 • Wilhelm, Maurer, Uebersetzerbau, 1997 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101501 Vorlesung Grundlagen des Compilerbaus und der Programmiersprachen • 101502 Übung Grundlagen des Compilerbaus und der Programmiersprachen 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Keine		

17b. Prüfungsleistungen:	Studienbegleitende Prüfung
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	10151 Grundlagen des Compilerbaus und der Programmiersprachen
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	<ul style="list-style-type: none">B.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog ISGB.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog ISWB.Sc. Softwaretechnik, 4. Semester<ul style="list-style-type: none">→ Ergänzungsmodule→ Katalog SWT

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:	Gunther Heidemann		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Gunther Heidemann 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3</p> <p>B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISW 1-3</p> <p>BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG</p>		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 080300100 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, Bildformate (z.B. ppm, gif, jpeg) • 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. AVI, MPEG) • Bildverbesserung und Restauration • Elementare Segmentierungsverfahren: Histogramme, Farben, Konturen • Elementare Mustererkennung <ul style="list-style-type: none"> • Fundamentals of optics such as pinhole camera and lens equation • Image acquisition: Cameras, lenses, illumination, acquisition process • Image representation: Discretization, color spaces, image file formats (e.g. ppm, gif, jpeg) • Basics of image processing, e.g. point operations such as contrast enhancement or binarization • Linear and nonlinear filtering such as convolution and morphological operations. 		

	<ul style="list-style-type: none"> • 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. avi, mpeg) • Image enhancement and restauration • Basics of segmentation: Histograms, colors, contours • Basics of pattern recognition
14. Literatur:	<ul style="list-style-type: none"> • Bässmann, Henning; Kreyss, 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 • Jähne, Bernd, Digitale Bildverarbeitung / Digital Image Processing, 2005 • 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. Abschaetzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden</p>
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	10171 Imaging Science
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	<p>B.Sc. Softwaretechnik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISW

Modul: 10250 Parallele Systeme

2. Modulkürzel:	051200065	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	Ein Fach 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:	<ul style="list-style-type: none"> • Literatur, siehe Webseite zur Veranstaltung 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102501 Vorlesung Parallele Systeme • 102502 Übung Parallele Systeme 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Keine		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10251 Parallele Systeme		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 3. Semester → Ergänzungsmodule → Katalog ISW		

Modul: 11110 Verteilte Systeme

2. Modulkürzel:	051200015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Kurt Rothermel		
9. Dozenten:	Kurt Rothermel		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051510005 Datenstrukturen und Algorithmen • 051200005 Systemkonzepte und -Programmierung 		
12. Lernziele:	<ul style="list-style-type: none"> • Verstehen der grundsätzlichen Eigenschaften, Konzepte und Verfahren verteilter Systeme. • Kann existierende verteilte Anwendungen und Systemplattformen hinsichtlich ihrer Eigenschaften analysieren und verstehen. • Kann verteilte Anwendungen/Systemplattformen auf der Grundlage der erlernten Methoden realisieren. • Kann sich mit Experten anderer Fachdisziplinen über die Anwendung verteilter Systeme verständigen. 		
13. Inhalt:	<ul style="list-style-type: none"> • Einführung in die verteilten Systeme • Systemmodelle • Kommunikation: Nachrichten, Remote Procedure Call (RPC), Remote Method Invocation RMI) • Namensgebung: Generierung und Resolution • Zeit und Uhren in verteilten Systemen: Anwendungen, logische Uhren, physikalische Uhren, Uhrensynchronisation • Prozesssynchronisation: Wechselseitiger Ausschluß • Globaler Zustand: Konzepte, Snapshot Algorithmus, verteiltes Debugging • Transaktionsmanagement: Serialisierbarkeit, Sperrverfahren, 2-Phasen-Commit-Protokolle • Datenreplikation: Primary Copy, Consensus-Protokolle und andere Algorithmen • Sicherheit: Verfahren zur Geheimhaltung, Integrität, Authentifikation und Autorisierung • Broadcast-Algorithmen: Verarbeitungsmodell, Broadcast-Semantiken und -Algorithmen 		
14. Literatur:	Literatur, siehe Webseite zur Veranstaltung		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 111101 Vorlesung Verteilte Systeme • 111102 Übungen Verteilte Systeme 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit:	42 h	
	Selbststudiumszeit / Nachbearbeitungszeit:	138 h	
	Gesamt:	180 h	

17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Schriftlich Prüfung von 90 Minuten Dauer (0.7) Benotete Übungen - studienbegleitend (0.3)
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 11111 Verteilte Systeme• 11112 Verteilte Systeme - Übungen
21. Angeboten von:	Institut für Parallele und Verteilte Systeme
22. Zuordnung zu weiteren Curricula:	<p>B.Sc. Softwaretechnik, 5. Semester → Ergänzungsmodule → Katalog ISG</p> <p>B.Sc. Wirtschaftsinformatik, 5. Semester → Informatik (B 1)</p> <p>B.Sc. Wirtschaftsinformatik, 5. Semester → Wahlpflichtbereich (Bereich C) → Wahlbereich II: Informatik Basis</p> <p>B.Sc. Wirtschaftsinformatik, 5. Semester → Wahlpflichtbereich (Bereich C) → Wahlbereich III: Informatik Simulation</p>

330 Katalog ISW 1-3

Zugeordnete Module: 10050 Bildverstehen
 10060 Computergraphik
 11900 Design and Test of Systems on a Chip
 10170 Imaging Science
 10180 Information Retrieval und Text Mining
 10250 Parallele Systeme
 10300 Rechnernetze

Modul: 10050 Bildverstehen

2. Modulkürzel:	051200035	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Paul Levi		
9. Dozenten:	<ul style="list-style-type: none"> • Paul Levi • Viktor Avrutin 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISW 1-3 <p>BA (Komb) Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Module im Nebenfach → Katalog ISW 		
11. Voraussetzungen:	Grundwissen über Programmierung, Datenstrukturen und Mathematik.		
12. Lernziele:	<p>Die Teilnehmer kennen und verstehen die Grundlagen der klassischen, verkörperten und verteilten Künstlichen Intelligenz. Ihnen wurden die dabei verwendeten Grundbegriffe so vorgestellt, dass sie als Bausteine von Architekturen intelligenter Systeme betrachtet werden. Dies bedeutet im Einzelnen, dass die Zuhörer am Beispiel des Bildverstehens die Wirkungsweise kognitiver Wahrnehmungsfähigkeiten kennengelernt haben, verstehen wie diese Fähigkeiten im Rahmen des Beobachtungs-, Planungs- und Aktionszyklus in einem Agenten eingesetzt werden und wie diese Fähigkeiten erweitert werden müssen, damit einzelne Agenten sich in einem Team kooperativ (Multiagenten-Systeme) verhalten können.</p> <p>Die Studierenden sind in der Lage, nicht nur einzelne wesentliche Algorithmen der Künstlichen Intelligenz (einschließlich des Bildverstehens) zu verstehen, sondern auch die wechselseitigen Beziehungen von verschiedenen Methoden zu berücksichtigen, um eine Beurteilung der Konzeption und der Wirkungsweise von intelligenten (kognitiven) technischen Systemen selbst vornehmen zu können.</p>		
13. Inhalt:	<ul style="list-style-type: none"> • Einleitung in Problemstellungen und Definitionen • Einführung in die Grundlagen der Bildverarbeitung • (Künstliche) Neuronale Netze • Bedingungsausbreitung (Constraints und ihre Propagierung) • Probabilistische Inferenz mit Bayes-Netzwerke • Verteilte Künstliche Intelligenz (VKI) • Multiagentensysteme (MAS) 		
14. Literatur:	<ul style="list-style-type: none"> • G. F. Luger, Künstliche Intelligenz, 2001 • Jähne, Bernd, Digitale Bildverarbeitung, 2005 • Literatur, siehe Webseite zur Veranstaltung • S. Russell, P. Norvig, Künstliche Intelligenz, 2004 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 100501 Vorlesung Bildverstehen • 100502 Übung Bildverstehen 		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	keine		

17b. Prüfungsleistungen: Schriftliche Prüfung von 90 Minuten Dauer.

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10051 Bildverstehen

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Softwaretechnik, 4. Semester
→ Ergänzungsmodule
→ Katalog ISW

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:	Daniel Weiskopf		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISW		
11. 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:	Folgende Themen werden in der Vorlesung behandelt: <ul style="list-style-type: none"> • Überblick über den Bildsyntheseprozess • Grundlegende Rastergraphik • 2D und 3D Geometrietransformationen, 3D Projektion • Polygonale und hierarchische Modelle • Verdeckungsrechnung • Grundlegende Renderingtechniken (Rasterung, Raytracing) • Beleuchtungsmodelle • Texturen • Grundlagen der geometrischen Modellierung (Kurven, Flächen) Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.		
14. Literatur:	<ul style="list-style-type: none"> • J. Encarnacao, W. Strasser, R. Klein, Graphische Datenverarbeitung (Band1 und 2), 1997 • J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics: Principle and Practice, 1990 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 100601 Vorlesung Computergraphik • 100602 Übung Computergraphik 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein.		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			

20. Prüfungsnummer/n und -name: 10061 Computergraphik

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Softwaretechnik, 5. Semester
→ Ergänzungsmodule
→ Katalog ISW
B.Sc. Maschinelle Sprachverarbeitung, 5. Semester
→ Wahlbereich E/I

Modul: 11900 Design and Test of Systems on a Chip

2. Modulkürzel:	051700015	5. Moduldauer:	-
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Melanie Elm 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISW		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051700005 Rechnerorganisation • Modul 051700010 Grundlagen der Rechnerarchitektur 		
12. Lernziele:	<p>The students of this course have gained a basic understanding of development and test of complex embedded hardware / software systems. The participants have become acquainted with the essential steps of synthesis, validation, test and programming and have learned, how to use the related tools for automation.</p> <p>Besides the different design styles, paradigms and standards the essential steps of automated design, test and programming of digital and mixed signal circuits have been discussed. Exercises and labs have served to practice the use of commercial tools and designs.</p>		
13. Inhalt:	<p>The course comprises:</p> <ul style="list-style-type: none"> • Overview over system design • Reuse and cores • Standards and platforms • Elements of analog and mixed signal designs • Design validation and verification • Test and design for testability with the related standards • Application and programming of embedded processors 		
14. Literatur:	<ul style="list-style-type: none"> • A. Sloss, D. Symes, C. Wright, ARM System Developer's Guide: Designing and Optimizing System Software, 2004 • L.-T. Wang, C.-W. Wu, X. Wen, VLSI Test Principles and Architectures - Design for Testability, 2006 • M. Keating, P. Bricaud, Reuse Methodology Manual for System-on-a-Chip Designs, 2007 • M. L. Bushnell, V. D. Agrawal, Essentials of Electronic Testing, 2005 • S. Furber, ARM System-on-Chip Architecture, 2000 • W. Wolf, Modern VLSI Design: System-on-Chip Design, 2002 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 119001 Vorlesung Design and Test of Systems on a Chip • 119002 Übung Design and Test of Systems on a Chip • 119003 Praktikum Design and Test of Systems on a Chip 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		

17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Lehrveranstaltungsbegleitende Prüfung (0.30) Schriftliche Prüfung von 90 Minuten Dauer (0.70)
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	11901 Design and Test of Systems on a Chip
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	

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:	Gunther Heidemann		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Gunther Heidemann 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3</p> <p>B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISW 1-3</p> <p>BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG</p>		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 080300100 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, Bildformate (z.B. ppm, gif, jpeg) • 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. AVI, MPEG) • Bildverbesserung und Restauration • Elementare Segmentierungsverfahren: Histogramme, Farben, Konturen • Elementare Mustererkennung <ul style="list-style-type: none"> • Fundamentals of optics such as pinhole camera and lens equation • Image acquisition: Cameras, lenses, illumination, acquisition process • Image representation: Discretization, color spaces, image file formats (e.g. ppm, gif, jpeg) • 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. avi, mpeg)
- Image enhancement and restoration
- Basics of segmentation: Histograms, colors, contours
- Basics of pattern recognition

14. Literatur:	<ul style="list-style-type: none"> • Bässmann, Henning; Kreyss, 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 • Jähne, Bernd, Digitale Bildverarbeitung / Digital Image Processing, 2005 • 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. Abschaetzung Arbeitsaufwand:	<p>Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden</p>
17a. Studienleistung:	<p>Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben</p>
17b. Prüfungsleistungen:	<p>Schriftliche Prüfung von 120 Minuten Dauer</p>
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<p>10171 Imaging Science</p>
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	<p>B.Sc. Softwaretechnik, 4. Semester → Ergänzungsmodule → Katalog ISW</p>

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:	Hinrich Schütze		
9. Dozenten:	<ul style="list-style-type: none"> • Helmut Schmid • Hinrich Schütze 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISW		
11. 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17a. Studienleistung:	regelmäßige Hausübungen		
17b. Prüfungsleistungen:	Klausur, Gewicht 1,0, schriftlich, 60 Minuten		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10181 Information Retrieval und Text Mining		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 5. Semester → Ergänzungsmodule → Katalog ISW B.Sc. Maschinelle Sprachverarbeitung, 5. Semester → Kernmodule		

Modul: 10250 Parallele Systeme

2. Modulkürzel:	051200065	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISG 1-3 B.Sc. Informatik, 3. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 5. Semester → Module im Nebenfach → Katalog ISG		
11. Voraussetzungen:	Ein Fach 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:	<ul style="list-style-type: none"> • Literatur, siehe Webseite zur Veranstaltung 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 102501 Vorlesung Parallele Systeme • 102502 Übung Parallele Systeme 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Keine		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10251 Parallele Systeme		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 3. Semester → Ergänzungsmodule → Katalog ISW		

Modul: 10300 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:	Kurt Rothermel		
9. Dozenten:	Kurt Rothermel		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Katalog ISW 1-3 BA (Komb) Informatik, 4. Semester → Module im Nebenfach → Katalog ISW		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051520005 Programmierung und Software-Entwicklung • 051510005 Datenstrukturen und Algorithmen 		
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 • L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach, 1999 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 103001 VL Rechnernetze • 103002 ÜB Rechnernetze 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden		

	Nachbearbeitungszeit: 138 Stunden
17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer (0.7) Benotete Übungen - studienbegleitend (0.3)
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 10301 Rechnernetze• 10302 Rechnernetze - Übungen
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 4. Semester → Ergänzungsmodule → Katalog ISW

340 Katalog ISW 4-7

Zugeordnete Module: 10160 Hardwarebeschreibungssprachen

Modul: 10160 Hardwarebeschreibungssprachen

2. Modulkürzel:	051711015	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Martin Radetzki		
9. Dozenten:	<ul style="list-style-type: none"> • Martin Radetzki • Sven Simon 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Katalog ISW 4-7 <p>BA (Komb) Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Module im Nebenfach → Katalog ISW 		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051711005 Technische Grundlagen der Informatik <p>oder</p> <ul style="list-style-type: none"> • Modul Einführung in die Technische Informatik (aus Studiengang Softwaretechnik, Modulnummer wird nachgetragen) 		
12. Lernziele:	<ul style="list-style-type: none"> • Kenntnis Hardwarebeschreibungssprachen • Kenntnis VHDL 		
13. Inhalt:	<p>Hardwarebeschreibungssprachen werden eingesetzt, um digitale Schaltungen auf der Algorithmen-, Registertransfer- und Gatterebene zu beschreiben. Sie ermöglichen eine ereignisgesteuerte Simulation und stellen Konzepte zur Beschreibung von Hierarchie, Nebenläufigkeit und Zeit zusätzlich zu den von Programmiersprachen bekannten Eigenschaften zur Verfügung. Dieses Modul vermittelt die wichtigsten Sprachkonstrukte der Hardwarebeschreibungssprache VHDL und ihre sinnvolle Anwendung zur Beschreibung digitaler Schaltungen und Systeme, indem die folgenden Themen vorgestellt und in Rechnerübungen angewandt werden:</p> <ul style="list-style-type: none"> • Entwurfshierarchie: Entities, Architectures, Instanzen, Verbindungen • Quellcodedateien und ihre Kompilation in Entwurfsbibliotheken • Nebenläufige sequentielle Prozesse • Sequentielle Anweisungen in VHDL • Typsystem von VHDL • Beschreibung typischer Hardwarestrukturen • VHDL für die Hardware synthese, Synthesesemantik • Beschreibung regulärer und rekursiver Strukturen • Simulationsmechanismen • Testbenches and Handhabung von Dateien • Organisation VHDL-basierter Projekte 		
14. Literatur:	<ul style="list-style-type: none"> • P. J. Ashenden, The Designer's Guide to VHDL (2nd edition), 2002 • P. J. Ashenden, The Student's Guide to VHDL, 1998 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 101601 Vorlesung Hardwarebeschreibungssprachen • 101602 Übung Hardwarebeschreibungssprachen 		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden		

	Nachbearbeitungszeit: 69 Stunden
17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Bewertete Rechnerübungen - vorlesungsbegleitend (0.5) Schriftliche Prüfung von 60 Minuten Dauer (0.5)
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	<ul style="list-style-type: none">• 10161 Hardwarebeschreibungssprachen• 10162 Hardwarebeschreibungssprachen - Rechnerübungen
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	

350 Wahlmodule aus Master Informatik

Zugeordnete Module:

- 10040 Bildsynthese
- 29570 Computer Interface Technologien
- 29580 Datenkompression
- 29590 Digitaler Systementwurf I
- 29440 Geometrische Modellierung und Animation
- 14380 Hardware Verification and Quality Assessment
- 29640 Mikrocontroller
- 29500 Visual Computing
- 11330 Visualisierung

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:	Thomas Ertl		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum:	<p>B.Sc. Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Ergänzungsmodule → Wahlmodule aus Master Informatik <p>BA (Komb) Informatik, 4. Semester</p> <ul style="list-style-type: none"> → Module im Nebenfach → Katalog ISW 		
11. Voraussetzungen:	<ul style="list-style-type: none"> • Modul 051900002 Computergraphik • Modul 051900001 Mensch-Computer-Interaktion 		
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.</p> <p>Interaktive Verfahren nutzen spezielle Eigenschaften moderner Graphikhardware, um mit Hilfe mehrdimensionaler Texturen und anderer Rasterisierungsoperationen realistische Beleuchtungseffekte in Echtzeit zu generieren. Bild-basierte Verfahren verzichten auf eine geometrische Repräsentation der Szene und erzeugen neue Ansichten aus anderen aufgenommenen Bildern.</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 		
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 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 100401 Vorlesung Bildsynthese • 100402 Übung Bildsynthese 		

16. Abschaetzung Arbeitsaufwand: Präsenzzeit: 42 Stunden
Nachbearbeitungszeit: 138 Stunden

17a. Studienleistung: Prüfungsvorleistung: Übungsschein.

17b. Prüfungsleistungen: Mündliche Prüfung von 30 Minuten Dauer.

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10041 Bildsynthese

21. Angeboten von:

22. Zuordnung zu weiteren Curricula: B.Sc. Softwaretechnik, 6. Semester
→ Ergänzungsmodule
→ Katalog ISW

Modul: 29570 Computer Interface Technologien

2. Modulkürzel:	051230105	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17a. Studienleistung:	keine / none		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Minuten oder mündliche Prüfung von 30 Min. Written exam of 120 min. or oral exam of 30 min.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	29571 Computer Interface Technologien		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:			

Modul: 29580 Datenkompression

2. Modulkürzel:	051230110	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	unregelmäßig
4. SWS:	4.0	7. Sprache:	Nach Ankündigung
8. Modulverantwortlicher:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. Voraussetzungen:	Grundlagenkenntnisse im Fach Mathematik.		
12. Lernziele:	Die Studierenden haben die Konzepte der Datenkompression erlernt und haben Verständnis über verschiedene Algorithmen zur Datenkompression erworben. Sie können die behandelten Algorithmen implementieren und weiterentwickeln.		
13. Inhalt:	<ul style="list-style-type: none"> • Shannons Entropie-Definition • Huffman Codierung • Arithmetische Codierung • Universal Codes • Verlustlose u. verlustbehaftete Kompression • Bilddatenkompression • Wörterbuch-basierte Kompression 		
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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17a. Studienleistung:	Keine		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Min. oder mündliche Prüfung von 30 Min.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	29581 Datenkompression		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:			

Modul: 29590 Digitaler Systementwurf I

2. Modulkürzel:	051230120	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. 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:			
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17a. Studienleistung:	keine / none		
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Min. oder mündlichen von 30 Min. Written exam of 120 min. or oral exam of 30 min.		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	29591 Digitaler Systementwurf I		
21. Angeboten von:			
22. Zuordnung zu weiteren Curricula:			

Modul: 29440 Geometrische Modellierung und Animation

2. Modulkürzel:	051900010	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	4.0	7. Sprache:	Englisch
8. Modulverantwortlicher:	Daniel Weiskopf		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf • Carsten Dachsbacher 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. Voraussetzungen:	Modul 051900002 Computergraphik Modul 051900001 Mensch-Computer-Interaktion Modul 051240005 Numerik und Stochastik		
12. Lernziele:	Die Studierende kennen die theoretischen Grundlagen der geometrischen Modellierung und Animation und haben praktische Fähigkeiten in der Programmierung von Verfahren der geometrischen Modellierung und Animation sowie in der Benutzung von Modellierungs- und Animationssoftware wie Maya erlangt.		
13. Inhalt:	<p>Die Vorlesung behandelt Grundlagen und ausgewählte Verfahren der Szenenmodellierung und der Computeranimation. Dazu gehören Kenntnisse über Kurven- und Flächenrepräsentationen, wie sie in Animationspaketen zur Modellierung von Objekten, zur Beschreibung von dynamischen Verhalten von Parametern und zur Keyframe-Interpolation verwendet werden.</p> <p>Physikalisch-basierte Animation hingegen beschreibt Bewegung durch die kinematischen und dynamischen Gesetze der Mechanik. Anwendungen reichen von Partikelsystemen bis zur Simulation von mehrgliedrigen Modellen und Verformungen.</p> <p>Folgende Themen werde in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> • Kurvenbeschreibung und -modellierung (allg. Polynomkurven, Bezier-Kurven, B-Splines, NURBS) • Flächenmodellierung (Tensorproduktflächen, NURBS) • Unterteilungsschemata • Überblick über Animationstechniken • Keyframe-Animation • Physikalisch basierte Animation <p>Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen und Anwendungen von Modellierungs- und Animationswerkzeugen (wie Maya) sowie theoretische Themen.</p>		
14. Literatur:	<ul style="list-style-type: none"> • D. Eberly, 3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics, 2000 		

- G. Farin, Curves and Surfaces for CAGD: A Practical Guide, 2002
- R. Parent, Computer Animation: Algorithms and Techniques, 2002

15. Lehrveranstaltungen und -formen:	294401 Vorlesung mit Übungen Geometrische Modellierung und Animation
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein / Passed Exercises
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Min. oder mündlichen Prüfung von 30 Min. Written exam of 120 min. or oral exam of 30 min.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	29441 Geometrische Modellierung und Animation
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	

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:	Hans-Joachim Wunderlich		
9. Dozenten:	<ul style="list-style-type: none"> • Hans-Joachim Wunderlich • Michael Kochte 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik BA (Komb) Informatik, 4. Semester → Module im Nebenfach → Katalog ISW		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051700005 Rechnerorganisation • 051711005 Technische Grundlagen der Informatik 		
12. Lernziele:	<ul style="list-style-type: none"> • Basic knowledge of methodologies and algorithms of functional and formal verification, diagnosis, test and design for testability of integrated circuits • Application of tools for simulation, verification and test insertion 		
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 • S. Minato, Binary Decision Diagrams and Applications for VLSI CAD, 1996 • T. Kropf, Introduction to Formal Hardware Verification, 1999 • W. Kunz, D. Stoffel, Reasoning in Boolean Networks, 1997 		
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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		

17a. Studienleistung:	Keine
17b. Prüfungsleistungen:	Schriftliche Prüfung von 90 Minuten Dauer
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	14381 Hardware Verification and Quality Assessment
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 4. Semester → Ergänzungsmodule → Katalog ISW

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:	Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. Voraussetzungen:	Kenntnisse in mindestens einer Programmiersprache und in mindestens einem Fach aus dem Bereich dem Bereich der Technischen Informatik oder ähnlichen Fächern.		
12. Lernziele:	Studierende beherrschen die praktische Programmierung von Mikrocontrollern und kennen klassische Architekturen. <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>Als Microcontroller (auch μController, μC, 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.</p> <p>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.).</p> <p>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.</p> <p>Aus http://de.wikipedia.org/wiki/Mikrocontroller</p>		
14. Literatur:	<ul style="list-style-type: none"> • Jörg Wiegmann, 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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden
17a. Studienleistung:	keine / none
17b. Prüfungsleistungen:	Schriftliche Prüfung von 120 Min. oder mündlichen Prüfung von 30 Min. Written exam of 120 min. or oral exam of 30 min.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	29641 Mikrocontroller
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	

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:	Daniel Weiskopf		
9. Dozenten:	Carsten Dachsbacher		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 0. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik		
11. Voraussetzungen:	Modul 051900002 Computergraphik Modul 051900001 Mensch-Computer-Interaktion		
12. Lernziele:	Die Studierenden kennen die theoretischen Grundlagen des Visual Computing und haben praktische Fähigkeiten in der Programmierung von Techniken für Visual Computing erlangt. Die Studierenden kennen verschiedene Ansätze und Algorithmen, die für Visual Computing einsetzbar sind.		
13. Inhalt:	<p>Diese Vorlesung behandelt die digitale Verarbeitung visueller Information die sich auf Methoden der Computergrafik, algorithmischen Geometrie, Bildbearbeitung aber auch Computer Vision und maschinelles Lernen stützt. Die Vorlesung gliedert sich in fünf Themenbereiche: Grundlagen in Signalverarbeitung, Geometrie und Farblehre, Bildverarbeitung, Bildsynthese, Geometrirepräsentation und Computing Vision.</p> <p>In dieser Vorlesung werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none"> • Farbe und Licht • Abtastung, Sampling Theorem, Quantisierung • Transformationen, Kameraabbildungen, Projektionen • Bildtransformationen und -filterung, AuflösungsPyramiden, Wavelets, Kompression • Segmentierung, Optischer Fluss • Grafik-Pipeline, Beleuchtungsberechnung, Texturen, Grafik-Hardware • Beschreibung von Kurven und Flächen • Maschinelles Lernen 		
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. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Selbststudium: 138 Stunden		
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein / Passed Excercises		

Modul: 11330 Visualisierung

2. Modulkürzel:	051900011	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Daniel Weiskopf		
9. Dozenten:	<ul style="list-style-type: none"> • Thomas Ertl • Daniel Weiskopf 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 4. Semester → Ergänzungsmodule → Wahlmodule aus Master Informatik BA (Komb) Informatik, 6. Semester → Module im Nebenfach → Katalog ISW		
11. Voraussetzungen:	<ul style="list-style-type: none"> • 051900002 Computergraphik • 051900001 Mensch-Computer-Interaktion • 051240005 Numerik und Stochastik. 		
12. Lernziele:	Die Studierenden haben Wissen über Grundlagen, Algorithmen und Datenstrukturen für die Visualisierung sowie praktische Fähigkeiten durch die Arbeit mit Visualisierungssoftware erworben.		
13. Inhalt:	<p>Visualisierung behandelt alle Aspekte, die mit der visuellen Repräsentation von Daten aus wissenschaftlichen Experimenten, Simulationen, medizinischen Scannern, Datenbanken oder ähnlichen Datenquellen gewonnen werden, um zu einem tieferen Verständnis zu gelangen oder eine einfachere Darstellung komplexer Phänomene oder Sachverhalte zu erhalten. Um dieses Ziel zu erreichen, werden zum einen wohlbekannte Techniken aus dem Gebiet der interaktiven Computergraphik, zum anderen auch neu entwickelte Techniken angewendet.</p> <p>Entsprechend werden in dieser Vorlesung folgenden Themen behandelt:</p> <ul style="list-style-type: none"> • Einführung, Historie, Visualisierungspipeline • Datenakquise und -repräsentation (Abtasten, Rekonstruktion, Gitter, Datenstrukturen) • Wahrnehmungsaspekte • Grundlegende Konzepte visueller Abbildungen • Visualisierung von Skalarfeldern (Isoflächenextraktion, Volumenrendering) • Visualisierung von Vektorfelder (Teilchenverfolgung, texturbasierte Methoden, Topologie) • Tensorfelder, Multiattributdaten • Hochdimensionale Daten und Informationsvisualisierung 		
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 • H. Schumann, W. Müller, Visualisierung: Grundlagen und allgemeine Methoden, 2000 • K. Engel, M. Hadwiger, J. M. Kniss, C. Rezk-Salama, D. Weiskopf, Real-time Volume Graphics, 2006 		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> • 113301 Vorlesung Visualisierung • 113302 Übungen Visualisierung 		

16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden
17a. Studienleistung:	Prüfungsvorleistung: Übungsschein.
17b. Prüfungsleistungen:	Mündliche Prüfung von 30 Minuten Dauer.
18. Grundlage für ... :	
19. Medienform:	
20. Prüfungsnummer/n und -name:	11331 Visualisierung
21. Angeboten von:	
22. Zuordnung zu weiteren Curricula:	B.Sc. Softwaretechnik, 6. Semester → Ergänzungsmodule → Katalog ISW B.Sc. Maschinelle Sprachverarbeitung, 5. Semester → Wahlbereich E/I

400 Schlüsselqualifikationen fachaffin

Zugeordnete Module: 10290 Projekt-INF
 10320 Seminar-INF

Modul: 10290 Projekt-INF

2. Modulkürzel:	051900095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	6.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	4.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Thomas Ertl		
9. Dozenten:			
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Voraussetzungen:	<ul style="list-style-type: none"> Basismodule der Informatik. Darüber hinaus variabel je nach Projektanforderung. (Wird per Aushang bekannt gegeben.)		
12. Lernziele:	<p>Die Teilnehmer können ein forschungsorientiertes Projekt unter Anleitung initiieren und planen. Sie können dazu notwendige Projektpläne erstellen, diese überwachen und ggf. den Realitäten anpassen. Sie können erforderliche Software beschaffen oder selbst erstellen. Sie verfügen insbesondere über die folgenden generischen Kompetenzen (Schlüsselqualifikationen): Sie können in Teams an einem gemeinsamen Vorhaben arbeiten und ihre Beiträge den übergeordneten Erfordernissen anpassen.</p> <p>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 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.</p> <p>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.</p>		
14. Literatur:	<ul style="list-style-type: none"> Literatur, die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben. 		
15. Lehrveranstaltungen und -formen:	102901 Seminar Projekt		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 42 Stunden Nachbearbeitungszeit: 138 Stunden		
17a. Studienleistung:	Keine.		
17b. Prüfungsleistungen:	<ul style="list-style-type: none"> USL (Projektschein ohne Note - Scheinkriterien: Aktive Teilnahme an den regelmäßigen Treffen und ein Projektbericht). 		
18. Grundlage für ... :			
19. Medienform:			
20. Prüfungsnummer/n und -name:	10291 Projekt-INF		

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:
- B.Sc. Technikpädagogik, 5. Semester
 - Hauptfach Informatik
 - Kernmodule Informatik
 - B.Sc. Technikpädagogik, 5. Semester
 - Wahlpflichtfach
 - Informatik
 - M.Sc. Technikpädagogik, 5. Semester
 - Wahlpflichtfach B
 - Affines Wahlpflichtfach Informatik
 - Affines Wahlpflichtfach Informatik Kernmodule
-

Modul: 10320 Seminar-INF

2. Modulkürzel:	050420095	5. Moduldauer:	1 Semester
3. Leistungspunkte:	3.0 LP	6. Turnus:	jedes 2. Semester, SoSe
4. SWS:	2.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> • Dozenten der Informatik • Dozenten der Anorganischen Chemie 		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 5. Semester → Schlüsselqualifikationen fachaffin		
11. Voraussetzungen:	<p>Basismodule der Informatik, darüber hinaus variabel: Je nach dem gewählten.</p> <p>Seminarthema können Vorkenntnisse aus weiteren Vorlesungen benötigt werden.</p>		
12. Lernziele:	<p>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.</p>		
13. Inhalt:	<p>Variabel: Es werden Seminare zu diversen häufig aktuellen Themen angeboten.</p> <p>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.</p>		
14. Literatur:	<ul style="list-style-type: none"> • Literatur, die begleitende Literatur wird in der Veranstaltung und im Web bekannt gegeben. 		
15. Lehrveranstaltungen und -formen:	103201 Seminar		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17a. Studienleistung:	Keine		

17b. Prüfungsleistungen: (Seminarschein mit Note - Scheinkriterien sind in der Regel ein Vortrag, eine schriftliche Ausarbeitung, sowie die aktive Mitarbeit während der Seminarveranstaltung).

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name: 10321 Seminar-INF

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:

- B.Sc. Wirtschaftsinformatik, 6. Semester
 - Wahlpflichtbereich (Bereich C)
 - Wahlbereich II: Informatik Basis
- B.Sc. Technikpädagogik, 5. Semester
 - Hauptfach Informatik
 - Kernmodule Informatik
