



Otto-Friedrich Universität Bamberg

Modulhandbuch

**Bachelor Software Systems Science
(Studienbeginn Sommersemester 2015)**

Wirtschaftsinformatik und Angewandte Informatik

****Stand: 16.12.2014****

Module

AI-Sem1-B: Bachelorseminar 1 der Fächergruppen Angewandte Informatik und Informatik.....	9
AI-Sem2-B: Bachelorseminar 2 der Fächergruppen Angewandte Informatik und Informatik.....	11
DSG-AJP-B: Fortgeschrittene Java-Programmierung.....	13
DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software.....	15
DSG-EiRBS-B: Einführung in Rechner- und Betriebssysteme.....	18
DSG-IDistrSys: Introduction to Distributed Systems.....	21
DSG-PKS-B: Programmierung komplexer interagierender Systeme.....	23
DSG-Project-2-SoSySc-B: DSG Bachelorprojekt Software Systems Science.....	25
EESYS-IITP-B: Internationales IT-Projektmanagement.....	27
Gdl-GTI-B: Grundlagen der Theoretischen Informatik.....	29
Gdl-MfI-1: Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik).....	31
Gdl-NPP-B: Nichtprozedurale Programmierung.....	33
Gdl-PR2-B: Bachelorprojekt Software Systems Science.....	35
Gdl-SaV-B: Logik (Specification and Verification).....	37
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis.....	39
HCI-IS-B: Interaktive Systeme.....	41
HCI-KS-B: Kooperative Systeme.....	43
HCI-US: Ubiquitäre Systeme.....	45
IAI-WAI-B: Wissenschaftliches Arbeiten in der Informatik.....	47
KInf-GeoInf-B: Geoinformationssysteme.....	49
KInf-SemInf-M: Semantic Information Processing.....	51
KogSys-IA-B: Intelligente Agenten.....	53
KTR-Datkomm-B: Datenkommunikation.....	55
KTR-GIK-M: Grundbausteine der Internet-Kommunikation.....	59
KTR-MfI-2: Mathematik für Informatiker 2 (Lineare Algebra).....	62
KTR-SSSProj-B: KTR Bachelorprojekt Software Systems Science.....	64
Mathe-B-01 (BWL): Mathematik für Wirtschaftswissenschaftler I (BWL).....	67
MI-AuD-B: Algorithmen und Datenstrukturen.....	69
MI-LA-DatSchu-B: Grundlagen und Fallstudien zum Datenschutz.....	71

Inhaltsverzeichnis

MI-WebT-B: Web-Technologien.....	73
MOBI-CONF: Scientific Conference Management.....	75
MOBI-DSC: Data Streams and Complex Event Processing.....	76
MOBI-IMP-B: Implementation of Data Management Systems.....	77
MOBI-MSS-B: Mobility in Software Systems.....	78
MOBI-PRS-B: Bachelor Project Mobile Software Systems (SoSySc).....	79
SEDA-DMS-B: Datenmanagementsysteme.....	81
SEDA-EuU-B: Entrepreneurship und Unternehmensgründung.....	83
SEDA-PT-B: Methoden der Präsentation, Gesprächsführung und Diskussion.....	85
SEDA-TA-B: Technikfolgeabschätzung / -bewertung.....	86
SME-Phy-B: Physical Computing.....	88
Stat-B-01: Methoden der Statistik I.....	90
Stat-B-02: Methoden der Statistik II.....	92
SWT-FSA-B: Foundations of Software Analysis.....	94
SWT-FSE-B: Foundations of Software Engineering.....	96
SWT-IPC-B: Imperative Programming Using C.....	98
SWT-PCC-M: Principles of Compiler Construction.....	100
SWT-PR2-B: SWT Bachelorprojekt Software Systems Science.....	102
SWT-SSP-B: Soft Skills in IT-Projekten.....	105
SWT-SWL-B: Software Engineering Lab.....	107

Übersicht nach Modulgruppen

1) A1 Mathematische Grundlagen (Modulgruppe) ECTS: 27

Mathe-B-01 (BWL): Mathematik für Wirtschaftswissenschaftler I (BWL) (3,00 ECTS, WS, SS).....	67
Gdl-Mfl-1: Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (6,00 ECTS, WS, jährlich).....	31
KTR-Mfl-2: Mathematik für Informatiker 2 (Lineare Algebra) (6,00 ECTS, SS, jährlich).....	62
Stat-B-01: Methoden der Statistik I (6,00 ECTS, WS, SS).....	90
Stat-B-02: Methoden der Statistik II (6,00 ECTS, WS, SS).....	92

2) A2 Allgemeine Informatik Grundlagen (Modulgruppe) ECTS: 36

DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software (6,00 ECTS, WS, jährlich).....	15
DSG-EiRBS-B: Einführung in Rechner- und Betriebssysteme (6,00 ECTS, SS, jährlich).....	18
Gdl-GTI-B: Grundlagen der Theoretischen Informatik (6,00 ECTS, SS, jährlich).....	29
MI-AuD-B: Algorithmen und Datenstrukturen (6,00 ECTS, SS, jährlich).....	69
SWT-FSA-B: Foundations of Software Analysis (6,00 ECTS, WS, jährlich).....	94
SEDA-DMS-B: Datenmanagementsysteme (6,00 ECTS, SS, jährlich).....	81

3) A3 Programmierung und Softwaretechnik (Modulgruppe) ECTS: 21 - 42

A3 Pflicht- und Wahlpflichtbereich Programmierung und Softwaretechnik (ab SS2015)

a) A3P Pflichtbereich Grundlagen PuSWT (Pflichtbereich) ECTS: 18

A3P Grundlagen aus dem Bereich Programmierung und Softwaretechnik

SWT-FSE-B: Foundations of Software Engineering (6,00 ECTS, SS, jährlich).....	96
SWT-SWL-B: Software Engineering Lab (6,00 ECTS, WS, jährlich).....	107
Gdl-NPP-B: Nichtprozedurale Programmierung (6,00 ECTS, WS, jährlich).....	33

b) A3W-I Wahlpflichtbereich I Schwerpunkt PuSWE (Wahlpflichtbereich) ECTS: 3

A3WP-I Vertiefungsmöglichkeiten zum Bereich Programmierung und Softwaretechnik

DSG-AJP-B: Fortgeschrittene Java-Programmierung (3,00 ECTS, SS, jährlich).....	13
SWT-IPC-B: Imperative Programming Using C (3,00 ECTS, WS, jährlich).....	98

c) A3W-II Wahlpflichtbereich II Schwerpunkt PuSWE (Wahlpflichtbereich) ECTS: 0 - 21

A3WP-II Vertiefungsmöglichkeiten zum Bereich Programmierung und Softwaretechnik (Hier kann außer den folgenden Modulen auch das noch nicht gewählte Modul aus dem Wahlpflichtbereich A3WP-I eingebracht werden.)

SWT-PCC-M: Principles of Compiler Construction (6,00 ECTS, WS, jährlich).....	100
MOBI-IMP-B: Implementation of Data Management Systems (6,00 ECTS, WS, jährlich).....	77
KInf-SemInf-M: Semantic Information Processing (6,00 ECTS, WS, jährlich).....	51
MI-WebT-B: Web-Technologien (6,00 ECTS, SS, jährlich).....	73
HCI-IS-B: Interaktive Systeme (6,00 ECTS, WS, jährlich).....	41
SME-Phy-B: Physical Computing (6,00 ECTS, SS, jährlich).....	88

4) A4 Komplexe und Verteilte Systeme (Modulgruppe) ECTS: 21 - 42

A4 Pflicht- und Wahlpflichtbereich Komplexe und Verteilte Systeme (ab SS2015)

a) A4P Pflichtbereich Grundlagen KuVS (Pflichtbereich) ECTS: 21

A4P Grundlagen aus dem Bereich Komplexe und Verteilte Systeme

DSG-PKS-B: Programmierung komplexer interagierender Systeme (3,00 ECTS, WS, jährlich).....	23
KTR-Datkomm-B: Datenkommunikation (6,00 ECTS, WS, jährlich).....	55
DSG-IDistrSys: Introduction to Distributed Systems (6,00 ECTS, SS, jährlich).....	21
MOBI-MSS-B: Mobility in Software Systems (6,00 ECTS, WS, jährlich).....	78

b) A4W Wahlpflichtbereich Komplexe und Verteilte Systeme (Wahlpflichtbereich) ECTS: 0 - 21

A4W Vertiefungsmöglichkeiten zum Bereich Komplexe und Verteilte Systeme

Gdl-SaV-B: Logik (Specification and Verification) (6,00 ECTS, WS, jährlich).....	37
KTR-GIK-M: Grundbausteine der Internet-Kommunikation (6,00 ECTS, SS, jährlich).....	59
HCI-KS-B: Kooperative Systeme (6,00 ECTS, SS, jährlich).....	43
KogSys-IA-B: Intelligente Agenten (6,00 ECTS, SS, jährlich).....	53
KInf-GeoInf-B: Geoinformationssysteme (6,00 ECTS, SS, jährlich).....	49
HCI-US: Ubiquitäre Systeme (6,00 ECTS, WS, jährlich).....	45
MOBI-DSC: Data Streams and Complex Event Processing (6,00 ECTS, WS, jährlich).....	76

5) A5 Fachstudium Anwendungsfächer (Modulgruppe) ECTS: 12 - 20

Anwendungsfächer mit Bezug zu Software Systems Science sind Wahlpflichtmodule anderer Fakultäten im Umfang von insgesamt 12 bis 20 ECTS-Punkten aus Bachelornebenfächern gemäß des Anhangs der Allgemeinen Prüfungsordnung für Bachelor- und Masterstudiengänge der Fakultäten Geistes- und Kulturwissenschaften, Humanwissenschaften und Sozial- und Wirtschaftswissenschaften an der Otto-

Friedrich-Universität Bamberg (vgl. <http://www.uni-bamberg.de/?id=29722>). Module aus dem Nebenfach Angewandte Informatik sind nicht wählbar.

WICHTIG: Die Modulprüfungen einzelner Module eines gewählten Anwendungsfachs müssen inklusive aller erforderlichen zugehörigen Modulteilprüfungen gemäß der gültigen Bestimmungen des gewählten Anwendungsfachs abgelegt werden. Die von der anbietenden Fakultät in der Studien- und Fachprüfungsordnung und im jeweiligen Modulhandbuch festgelegten Regelungen zur Anmeldung und Anrechnung von Modulprüfungen einzelner Module und ihrer zugeordneten Lehrveranstaltungen sind bindend. Im Hinblick auf einzelne Anwendungsfächer gelten oft Zugangsvoraussetzungen, so sind zum Teil Vorpraktika erforderlich. Zu den Details können die Informationen des jeweiligen Anwendungsfachs und die Informationsangebote der dortigen Fachstudienberatung herangezogen werden, vgl: <http://www.uni-bamberg.de/?id=29722> . Mit der ersten Prüfungsanmeldung zu einem Modul verpflichtet sich der Studierende sowohl zur Beachtung der Zuordnungsregelungen von Lehrveranstaltungen und Modulteilprüfungen des gewählten Moduls als auch der Anerkennung der Prüfungsanmeldungs- und -anrechnungsregelungen des Moduls in dem gewählten Anwendungsfach.

6) A6 Kontextstudium (Modulgruppe) ECTS: 12 - 16

a) Allgemeine Schlüsselqualifikationen (Teil-Modulgruppe) ECTS: 0 - 16

Beachten Sie auch die Informationen des Prüfungsausschusses.

EESYS-IITP-B: Internationales IT-Projektmanagement (6,00 ECTS, SS, jährlich).....	27
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis (3,00 ECTS, SS, jährlich).....	39
MI-LA-DatSchu-B: Grundlagen und Fallstudien zum Datenschutz (4,00 ECTS, SS, jährlich).....	71
SEDA-PT-B: Methoden der Präsentation, Gesprächsführung und Diskussion (3,00 ECTS, WS, SS).....	85
SEDA-EuU-B: Entrepreneurship und Unternehmensgründung (3,00 ECTS, SS, jährlich).....	83
SWT-SSP-B: Soft Skills in IT-Projekten (3,00 ECTS, SS, jährlich).....	105
MOBI-CONF: Scientific Conference Management (3,00 ECTS, SS, jährlich).....	75

b) Wissenschaftliches Arbeiten (Teil-Modulgruppe) ECTS: 0 - 3

IAI-WAI-B: Wissenschaftliches Arbeiten in der Informatik (3,00 ECTS, SS, jährlich).....	47
---	----

c) Fremdsprachen (Wahlpflichtbereich) ECTS: 0 - 16

Module gemäß des aktuellen Angebots des Sprachenzentrums, insb. English for IT I und English for IT II.

d) Philosophie / Ethik (Teil-Modulgruppe) ECTS: 0 - 16

SEDA-TA-B: Technikfolgeabschätzung / -bewertung (3,00 ECTS, SS, jährlich).....	86
--	----

7) A7 Seminare und Projekte (Modulgruppe) ECTS: 18

a) Seminare (Wahlpflichtbereich) ECTS: 6

AI-Sem1-B: Bachelorseminar 1 der Fächergruppen Angewandte Informatik und Informatik (3,00 ECTS, WS, SS).....9

AI-Sem2-B: Bachelorseminar 2 der Fächergruppen Angewandte Informatik und Informatik (3,00 ECTS, WS, SS).....11

b) Projekte (Wahlpflichtbereich) ECTS: 12

DSG-Project-2-SoSySc-B: DSG Bachelorprojekt Software Systems Science (12,00 ECTS, WS, jährlich)..... 25

GdI-PR2-B: Bachelorprojekt Software Systems Science (12,00 ECTS, WS, SS)..... 35

KTR-SSSProj-B: KTR Bachelorprojekt Software Systems Science (12,00 ECTS, WS, SS)..... 64

SWT-PR2-B: SWT Bachelorprojekt Software Systems Science (12,00 ECTS, SS, jährlich).....102

MOBI-PRS-B: Bachelor Project Mobile Software Systems (SoSySc) (12,00 ECTS, WS, SS)..... 79

8) A8 Bachelorarbeit (Modulgruppe) ECTS: 12

Bachelorarbeit gemäß § 35 und Anhang 2 StuFPO B. Sc. Software Systems Science.

Modul AI-Sem1-B Bachelorseminar 1 der Fächergruppen Angewandte Informatik und Informatik		3 ECTS / 90 h
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Udo Krieger		
Inhalte: Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet mit wissenschaftlichen Methoden.		
Lernziele/Kompetenzen: Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
Bemerkung: Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik oder Informatik zu wählen.		
Empfohlene Vorkenntnisse: Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Bachelorseminar 1 Lehrformen: Seminar Sprache: Deutsch/Englisch	2,00 SWS
Lernziele: Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.	
Inhalte: Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
Literatur: Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.	

Prüfung Hausarbeit mit Referat Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung Beschreibung: Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.	
--	--

Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Referats werden zu Beginn jeder Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekanntgegeben.	
---	--

Modul AI-Sem2-B Bachelorseminar 2 der Fächergruppen Angewandte Informatik und Informatik		3 ECTS / 90 h
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Udo Krieger		
Inhalte: Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet mit wissenschaftlichen Methoden.		
Lernziele/Kompetenzen: Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
Bemerkung: Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik oder Informatik zu wählen.		
Empfohlene Vorkenntnisse: Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Bachelorseminar 2 Lehrformen: Seminar Sprache: Deutsch/Englisch	2,00 SWS
Lernziele: Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.	
Inhalte: Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
Literatur: Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.	

Prüfung Hausarbeit mit Referat Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung Beschreibung: Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.	
--	--

<p>Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Referats werden zu Beginn jeder Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekanntgegeben. Die Gewichtung der Prüfungsleistungen Hausarbeit und Referat wird zu Beginn der Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekanntgegeben.</p>	
---	--

Modul DSG-AJP-B Fortgeschrittene Java-Programmierung		3 ECTS / 90 h
<i>Advanced Java Programming</i>		
Version 4.0.0 (seit WS12/13) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
Inhalte: Aufbauend auf den Grundkenntnissen der objekt-orientierten Programmierung in Java aus DSG-EiAPS-B soll der Umgang mit modernen objekt-orientierten Programmiersprachen durch einen genaueren Blick auf die Möglichkeiten, die eine moderne Programmierumgebung heute liefert, vertieft und gefestigt. Dazu gehören als Themen - jeweils am Beispiel 'Java' praktisch erläutert und geübt - insbesondere: <ul style="list-style-type: none"> • Interfaces, abstrakte Klassen und komplexere Vererbungsstrukturen, Nutzung von Package-Strukturen, • Einsatz und Behandlung von Exceptions, • Nutzung komplexer Java-APIs, z.B. für Ein- und Ausgabe, • grundlegende XML Verarbeitung, • Debugging, Profiling und Testen, • Überblick über das Programmieren von (grafischen) Benutzerschnittstellen (G)UIs. Zusätzlich werden die ersten Schritte zur Nutzung komplexer Programmierumgebungen, die über den einfachen Editor-Compiler-Ausführungs-Zyklus hinausgehen, insbesondere der Umgang mit einfachen Testszenarien zur Entwicklung verlässlicher Systeme, eingeübt.		
Lernziele/Kompetenzen: Die Studierenden beherrschen die grundlegenden Mechanismen der objekt-orientierten Programmierung vertieft und sind auch in der Lage, einfache Probleme mit Hilfe der über die Standardprogrammiersprachen-Konstrukte hinausgehenden Hilfsmittel einer modernen Programmierumgebung effizient und flexibel zu lösen.		
Bemerkung: Der Arbeitsaufwand von 90 Std. gliedert sich in <ul style="list-style-type: none"> • 22.5 Std. Teilnahme an der Praktischen Übung • 55 Std. Bearbeiten von Programmieraufgaben/assignments • 12 Std. Vorbereitung auf das Abschlusskolloquium • 0.5 Std. Abschlusskolloquium inklusive Warten auf Ergebnis usw. 		
Empfohlene Vorkenntnisse: Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Algorithmik und Softwareentwicklung, wie sie z.B. im Modul DSG-EiAPS-B vermittelt werden. Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester: 2.	Minimale Dauer des Moduls: 1 Semester
Lehrveranstaltungen		
Praktische Übung Fortgeschrittene Java-Programmierung		2,00 SWS

<p>Lehrformen: Vorlesung und Übung Dozenten: Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik Sprache: Deutsch</p>	
<p>Inhalte: vgl. Modulbeschreibung</p>	
<p>Literatur: Jedes weiterführende Buch zu Java ist verwendbar.</p>	
<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten Bearbeitungsfrist: 3 Monate Beschreibung: Während des Semesters regelmäßig ausgegebene Programmieraufgaben (Assignments) werden als Vorbereitung auf das Abschlusskolloquium von den Studierenden gelöst; zu den Lösungen gibt es inhaltliches Feedback und Hilfestellungen von den betreuenden Mitarbeitern schon während des Semesters. Im Abschlusskolloquium stellen die Studierenden (jeweils einzeln) die von ihnen während des Semesters erarbeiteten Lösungen zu den semesterbegleitend ausgegebenen Programmieraufgaben vor, erläutern diese und beantworten Fragen zu den vorgestellten Lösungen und den dabei verwendeten Techniken und Sprachkonstrukten.</p>	

Modul DSG-EiAPS-B Einführung in Algorithmen, Programmierung und Software <i>Introduction to Algorithms, Programming and Software</i>	6 ECTS / 180 h
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Guido Wirtz	
<p>Inhalte:</p> <p>Das Modul gibt einen ersten Einblick in die Informatik aus Sicht der Entwicklung von Algorithmen und deren Realisierung durch Programme in einer imperativen, objekt-orientierten Programmiersprache (am Beispiel von Java) sowie einen Ausblick auf die Problematik der Softwareentwicklung. Behandelt werden die Grundprinzipien der Informatik zu:</p> <ul style="list-style-type: none"> • Präsentation, Interpretation und Manipulation von Information, • Syntax und Semantik von einfachen Sprachen, • Probleme, Problemklassen und -Instanzen, • Design, Entwicklung und Implementierung von Algorithmen für einfache Problemklassen, • einfache Datenstrukturen wie Keller, Warteschlangen, Listen und Bäume, sowie • Techniken zur Spezifikation, zur Datenabstraktion und funktionalen Abstraktion. <p>All diese Begriffe werden am Beispiel der Programmiersprache 'Java' diskutiert, so dass auch die wesentlichen Konzepte imperativer und objekt-orientierter Programmiersprachen wie</p> <ul style="list-style-type: none"> • Wertebereiche, Namensräume, Speichermodelle und Zuweisungen, • Kontroll- und Datenfluss in einem Programm, • Iteration und Rekursion, sowie • Klassen, Schnittstellen, Vererbung, Polymorphie und Fehlerbehandlung <p>besprochen und auch praktisch eingeübt werden.</p>	
<p>Lernziele/Kompetenzen:</p> <p>Studierende haben einen ersten Überblick über das Fach 'Informatik' mit seinen verschiedenen Gebieten und kennen die grundlegenden Begriffe, Methoden und Techniken der Informatik aus Sicht von Algorithmen, Programmiersprachen und Softwareentwicklung. Studierende sind in der Lage, geeignete Abstraktions- und Repräsentationsmethoden zur maschinellen Bearbeitung auszuwählen und Methoden zur Beschreibung von Syntax und Semantik einfacher Sprachen anzuwenden. Sie verstehen die Zusammenhänge zwischen Spezifikation und Implementierung wie auch die Arbeitsweise einer Programmiersprache und können die wesentlichen Schritte der Softwareentwicklung nachzuvollziehen. Studierende können einfache Problemstellungen beschreiben, algorithmische Lösungen dazu entwickeln und diese auch in Java mittels einfacher Datenstrukturen umsetzen.</p>	
<p>Bemerkung:</p> <p>Der Arbeitsaufwand von 180 Std. verteilt sich ausgehend von einem 15 Arbeitswochen dauernden Semester in etwa wie folgt:</p> <ul style="list-style-type: none"> • 22.5 Std. Vorlesungsteilnahme • 22.5 Std. Übungsteilnahme • 60 Std. Bearbeiten von wöchentlichen Übungsaufgaben (d.h. ca. 4 Std./Woche) • 30 Std. Vor- und Nachbereitung (Literatur, Recherchen usw.) von Vorlesung und Übung (d.h. ca. 1.5 Std./Woche ohne Bearbeiten der Übungsaufgaben) 	

<ul style="list-style-type: none"> • 45 Std. Vorbereitung auf und Zeit für die Abschlussklausur (unter Annahme der o.g. Arbeitsaufwände während des Semesters) <p>Bei diesen Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht noch wird die regelmäßige Bearbeitung von Aufgaben formal überprüft. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.</p>		
Empfohlene Vorkenntnisse: Die Veranstaltung hat als grundlegende Einführungsveranstaltung in das Gebiet der Informatik weder Inhalte anderer Lehrveranstaltungen noch Informatikkenntnisse oder Programmierkenntnisse zur Voraussetzung. <i>Insbesondere ist das Modul DSG-EiRBS-B, das regelmäßig im Sommersemester angeboten wird, keine Voraussetzung für DSG-EiAPS-B.</i>		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. DSG-EiAPS-B: Vorlesung Einführung in Algorithmen, Programmierung und Software Lehrformen: Vorlesung Dozenten: Prof. Dr. Guido Wirtz Sprache: Deutsch <hr/> Inhalte: vgl. Modulbeschreibung <hr/> Literatur: Jede Einführung in die Informatik oder in die Programmiersprache Java kann als Ergänzung zur Veranstaltung genutzt werden, allerdings orientiert sich die Vorlesung nicht an einem Buch; deshalb ist die Liste hier nur als Auswahl "nützlicher" Bücher zu verstehen: <ul style="list-style-type: none"> • Heinz-Peter Gumm, Manfred Sommer: Einführung in die Informatik. Oldenbourg Verlag, 2011 (9th) • Barbara Liskov with John Guttag: Program Development in Java. Addison-Wesley, 2001 • Timothy Budd: An Introduction to Object-Oriented Programming, Pearson/ Addison Wesley, 2002 (3rd) • Christian Ullenboom: Java ist auch eine Insel. Galileo Computing, 2012 (10th) • John Lewis, Joseph Chase: Java Software Structures. Pearson/ Addison-Wesley, 2010 (3rd) 	2,00 SWS
2. DSG-EiAPS-B Übung Lehrformen: Übung Dozenten: Mitarbeiter Praktische Informatik Sprache: Deutsch <hr/> Inhalte:	2,00 SWS

In der Übung werden die wichtigsten Konzepte der gleichnamigen Vorlesung an einfachen Beispielen praktisch umgesetzt und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema, die den Studierenden regelmäßig zum freiwilligen Üben angeboten werden, vertieft. Dabei wird insbesondere Wert auf die Vorstellung von Lösungen durch die Studierenden und deren Diskussion in der Übungsgruppe gelegt. Im Rahmen der Übungen finden auch Rechnerübungen zum Thema 'Einführung in Java und die Java-Umgebung' in den Rechnerpools der Fakultät statt, die insbesondere Programmieranfängerinnen und -anfänger den Einstieg durch vor Ort Hilfe erleichtern sollen.

Literatur:

vgl. Vorlesung

Prüfung

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

Beschreibung:

Klausur zum Stoff des gesamten Moduls, also der Vorlesung, Übung und Rechnerübung zur DSG-EiAPS-B.

<p>Modul DSG-EiRBS-B Einführung in Rechner- und Betriebssysteme <i>Introduction to Computer Architecture and Operating Systems</i></p>	<p>6 ECTS / 180 h</p>
<p>Version 1.0.0 (seit SS14) Modulverantwortliche/r: Prof. Dr. Guido Wirtz</p>	
<p>Inhalte: Die Modul bietet einen ersten Einblick in die Informatik der Systeme. Neben einer an Systemen ausgerichteten Einführung in die Informatik behandelt die Veranstaltung die Aufgaben und Architekturmerkmale von Rechner- und Betriebssystemen. Sie bietet einen Einblick in Aufbau und Architektur monolithischer Rechnersysteme. Dazu gehört neben dem schrittweisen Aufbau eines minimalen Rechners, beginnend mit aussagenlogischen Ausdrücken über ihre Realisierung durch Gatter und Standardbausteine sowie zustandsbehaftete Schaltungen und Speicherbausteinen auch die Darstellung von Daten im Rechner und ihre detaillierte Speicherung und Verarbeitung. Zusätzlich wird ein Überblick über das Zusammenspiel von Konzepten der Rechnerarchitektur mit den wichtigsten Prinzipien und Komponenten von Systemsoftware (Prozess- und Ressource-Scheduling, Speicherverwaltung, Hintergrundspeicher, I/O-Handhabung) gegeben. Die Vorlesung gibt zusätzlich einen Ausblick auf moderne Techniken der Prozessorarchitektur und Multiprozessorarchitekturen, wie sie in aktuellen Serverkonstellationen zum Einsatz kommen. Die Themen werden anhand von Modellen sowie anhand von marktgängigen Rechner- und Betriebssystemen behandelt.</p> <p>Bemerkung: In diesem Modul wird bewusst vollständig auf die Vermittlung von Programmierkenntnissen verzichtet.</p>	
<p>Lernziele/Kompetenzen: Studierende haben einen ersten Überblick über die verschiedenen Gebiete der Informatik und kennen die grundlegenden Begriffe und Methoden der Informatik wie die wichtigsten in der Informatik verwendeten Techniken sowohl aus Sicht der 'Informatik der Systeme'. Die Studierenden haben ein grundlegendes Verständnis zustandsbasierter Systeme und der darin möglichen Abläufe (Prozesse). Zusätzlich kennen sie den Aufbau moderner Rechner- und Betriebssysteme und die dabei zur Anwendung kommenden Informatiktechniken.</p>	
<p>Bemerkung: Der Arbeitsaufwand von 180 Std. verteilt sich ausgehend von einem 15 Arbeitswochen dauernden Semester in etwa wie folgt:</p> <ul style="list-style-type: none"> • 22.5 Std. Vorlesungsteilnahme • 22.5 Std. Übungsteilnahme • 60 Std. Bearbeiten von wöchentlichen Übungsaufgaben (d.h. ca. 4 Std./Woche) • 30 Std. Vor- und Nachbereitung (Literatur, Recherchen usw.) von Vorlesung und Übung (d.h. ca. 1.5 Std./Woche ohne Bearbeiten der Übungsaufgaben) • 45 Std. Vorbereitung auf und Zeit für die Abschlussklausur (unter Annahme der o.g. Arbeitsaufwände während des Semesters) <p>Bei diesen Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht noch wird die regelmäßige Bearbeitung von Aufgaben formal überprüft. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.</p>	
<p>Empfohlene Vorkenntnisse:</p>	<p>Besondere Bestehensvoraussetzungen:</p>

Die Veranstaltung hat als grundlegende Einführungsveranstaltung in das Gebiet der Informatik der Systeme weder Inhalte anderer Lehrveranstaltungen noch Informatikkenntnisse oder Programmierkenntnisse zur Voraussetzung. <i>Insbesondere ist das Modul DSG-EiAPS-B, das regelmäßig im Wintersemester angeboten wird, keine Voraussetzung für DSG-EiRBS-B.</i>		Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
<p>1. DSG-EiRBS-B: Vorlesung Einführung in Rechner- und Betriebssysteme Lehrformen: Vorlesung Dozenten: Prof. Dr. Guido Wirtz Sprache: Deutsch</p> <p>Inhalte: vgl. Modulbeschreibung</p> <p>Literatur: Zum Bereich Rechnerarchitektur und Betriebssysteme gibt es eine ganze Reihe guter einführender Bücher, die aber alle über den in der Vorlesung behandelten Stoff hinausgehen. Deshalb ist die folgende Liste nur als Hinweis auf ergänzende Literatur gedacht - die Veranstaltung kann auch ohne auch nur eins dieser Bücher erfolgreich absolviert werden. Zu Beginn des Semesters wird zudem ein vollständiges, ausführliches Skript elektronisch zur Verfügung gestellt.</p> <ul style="list-style-type: none"> • Tanenbaum, A.S./Austin, T.: Structured Computer Organization. Addison-Wesley, 2012 (6th) • Murdocca, M./Heuring, V.P.: Computer Architecture and Organization. Prentice Hall 2007 (1th) • Tanenbaum, A.S.: Moderne Betriebssysteme. Pearson Studium 2009 (3rd) • Silberschatz, A./Gagne, G./Galvin, P B.: Operating Systems Concepts. John Wiley and Sons, 2012 (9th) 	2,00 SWS
<p>2. DSG-EiRBS-B Übung Lehrformen: Übung Dozenten: Mitarbeiter Praktische Informatik Sprache: Deutsch</p> <p>Inhalte: In der Übung werden die wichtigsten Konzepte der gleichnamigen Vorlesung an einfachen Beispielen praktisch umgesetzt und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema, die den Studierenden regelmäßig zum freiwilligen Üben angeboten werden, vertieft. Dabei wird insbesondere Wert auf die Vorstellung von Lösungen durch die Studierenden und deren Diskussion in der Übungsgruppe gelegt.</p> <p>Literatur: vgl. Vorlesung</p>	2,00 SWS

<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: Klausur zum Stoff des gesamten Moduls, also der Vorlesung und Übung zur DSG-EiRBS-B.</p>	
---	--

Modul DSG-IDistrSys Introduction to Distributed Systems		6 ECTS / 180 h
<i>Introduction to Distributed Systems</i>		
Version 1.0.0 (seit SS13) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
Inhalte: The course introduces to the different flavors of and issues with distributed systems, discusses the most basic problems arising with this kind of systems and presents solutions and techniques that are essential to make distributed systems work. Additionally, the course also teaches how to build simple distributed systems using Java-based technologies like process interaction, synchronization, remote message invocation and web service infrastructure.		
Lernziele/Kompetenzen: Students know about the characteristics and different flavors of distributed systems and understand the essential differences compared to monolithic, centralized systems as well as their consequences when designing and building distributed systems. Students are able to apply the basic algorithmic techniques and programming paradigms in order to build simple distributed systems themselves. Students have gained basic experience with practically building and running distributed systems.		
Bemerkung: The language of instruction in this course is English. The overall workload of 180h for this module consists of: <ul style="list-style-type: none"> • weekly classes: 22.5h • tutorials: 22.5h • Work on assignments: 75h • Literature study 30h • preparation for and time of the final exam: 30h This course is intended for 2nd/3rd year bachelor students as well as master students which have not enrolled in a similar course during their bachelor studies. In case of questions don't hesitate to contact the person responsible for this module.		
Empfohlene Vorkenntnisse: Knowledge of the basics of computer science in general, esp. operating systems, as well as practical experience in Java programming, as the subjects taught in DSG-EiAPS-B and DSG-EiRBS-B. Preferable also knowledge about multithreading and synchronization like, e.g., the subject-matters of DSG-PKS-B. Modul Programmierung komplexer interagierender Systeme (DSG-PKS-B) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
Lehrveranstaltungen		
1. Lecture Introduction to Distributed Systems Lehrformen: Vorlesung		2,00 SWS

<p>Dozenten: Prof. Dr. Guido Wirtz Sprache: Englisch</p> <hr/> <p>Inhalte: c.f. overall module description</p> <hr/> <p>Literatur:</p> <ul style="list-style-type: none"> • George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems. Pearson Education UK, 2011 (5. Auflage); ISBN: 9780273760597 • Kenneth P. Birman: Guide to Reliable Distributed Systems. Springer Texts in CS, Springer Verlag, 2012, ISBN 978-1-4471-2415-3 	
<p>2. Tutorial Introduction to Distributed Systems</p> <p>Lehrformen: Übung</p> <p>Dozenten: Mitarbeiter Praktische Informatik</p> <p>Sprache: Englisch/Deutsch</p> <hr/> <p>Inhalte: Introduction to and discussion of tools and practical issues closely related to the topics discussed in the lecture as well as solutions of problems that come up during working on the practical assignments.</p>	<p>2,00 SWS</p>
<p>Prüfung</p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten Bearbeitungsfrist: 3 Monate</p> <p>Beschreibung: Oral examination concerning the topics discussed in the lecture, exercises and assignments.</p>	

Modul DSG-PKS-B Programmierung komplexer interagierender Systeme <i>Introduction to Parallel and Distributed Programming</i>		3 ECTS / 90 h
Version 3.0.0 (seit WS12/13) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<p>Inhalte:</p> <p>Die Veranstaltung erläutert und übt den Umgang mit (explizit) parallelen Programmen und schafft damit auch ein vertieftes Verständnis für die Arbeitsweise heutiger Mehrkernprozessoren und Multiprozessoren. Dabei wird sowohl auf die grundlegenden Probleme und Techniken eingegangen als auch das praktische Entwerfen und Programmieren solcher Systeme (derzeit auf der Grundlage von Java) eingeübt. Dabei geht es um</p> <ul style="list-style-type: none"> • Prozesse und Threads, • Prozesskommunikation, • Synchronisation bei Shared Memory, • einfache C/S-Systeme mit TCP sockets, • Message-Passing im Aktor-Modell. <p>Zusätzlich wird die Problematik robuster verteilter Systeme diskutiert und ein Ausblick auf alternative Interaktionsparadigmen gegeben.</p>		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden kennen die gebräuchlichen Prozessbegriffe, die grundsätzlichen Probleme der Programmierung echt- und pseudo-paralleler Prozesssysteme sowie die grundlegenden Mechanismen zur Inter-Prozess-Kommunikation. Die Studierenden sind in der Lage, einfache parallele Programme mittels Threads zu schreiben, diese über Synchronisationsverfahren zu koordinieren sowie durch Kommunikationsmechanismen kooperativ und verlässlich zusammen arbeiten zu lassen.</p>		
<p>Bemerkung:</p> <p>Der Arbeitsaufwand von 90 Std. gliedert sich in</p> <ul style="list-style-type: none"> • 22.5 Std. Teilnahme an der Praktischen Übung • 55 Std. Bearbeiten von Programmieraufgaben/assignments • 12 Std. Vorbereitung auf das Abschlusskolloquium • 0.5 Std. Abschlusskolloquium inkl. Warten auf ergebnis usw. 		
<p>Empfohlene Vorkenntnisse:</p> <p>Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Betriebssysteme, wie sie z.B. im Modul DSG-EiRBS-B vermittelt werden.</p> <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen</p> <p>Modul Einführung in Rechner- und Betriebssysteme (DSG-EiRBS-B) - empfohlen</p>		<p>Besondere Bestehensvoraussetzungen:</p> <p>Siehe Prüfungsordnung</p>
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
<p>Praktische Übung Programmierung komplexer interagierender Systeme Lehrformen: Vorlesung und Übung Dozenten: Mitarbeiter Praktische Informatik Sprache: Deutsch</p>	2,00 SWS
<p>Inhalte: vgl. Modulbeschreibung</p>	
<p>Literatur: - wird jeweils aktuell zur Veranstaltung angegeben -</p>	
<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten Bearbeitungsfrist: 3 Monate Beschreibung: Während des Semesters regelmäßig ausgegebene Programmieraufgaben (Assignments) werden als Vorbereitung auf das Abschlusskolloquium von den Studierenden gelöst; zu den Lösungen gibt es inhaltliches Feedback und Hilfestellungen von den betreuenden Mitarbeitern schon während des Semesters. Im Abschlusskolloquium stellen die Studierenden (jeweils einzeln) die von ihnen während des Semesters erarbeiteten Lösungen zu den semesterbegleitend ausgegebenen Programmieraufgaben vor, erläutern diese und beantworten Fragen zu den vorgestellten Lösungen und den dabei verwendeten Techniken und Sprachkonstrukten.</p>	

Modul DSG-Project-2-SoSySc-B DSG Bachelorprojekt Software Systems Science <i>DSG Bachelorproject Software Systems Science</i>	12 ECTS / 360 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Guido Wirtz	
Inhalte: Überschaubare Themen aus der aktuellen Forschungsarbeit der Arbeitsgruppe Verteilte Systeme (DSG), die aber eine umfangreiche Einarbeitung erfordern können, werden in einer zum Teil gemeinsam, zum Teil arbeitsteilig, arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung im Rahmen des zweisemestrigen Projekts durchgeführt. Dabei geht es nicht nur um die programmiertechnische Umsetzung, sondern insbesondere auch um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe, sowie um die Sicherstellung der robusten und verlässlichen Funktion der entwickelten Systeme. In der Regel wird dazu das Studium aktueller Literatur und die Auswahl, Umsetzung und/oder Adaption zum Thema vorgeschlagener Ansätze notwendig sein. Typische Themen - die sich jeweils den aktuellen Arbeiten der DSG anpassen - sind z.B. die Untersuchung von BPMN- oder BPEL basierten Standards und Ansätzen im Bereich von dienst-orientierten Systemen oder aber die Erstellung eines Prototyps zum Monitoring oder der Visualisierung verteilter Software-Systeme.	
Lernziele/Kompetenzen: Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der Verteilten Systeme geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht sowie einer Posterpräsentation.	
Bemerkung: Dieses Modul erstreckt sich über 2 Semester (Start im Wintersemester): 2x6=12 ECTS, 2x4=8 SWS. Der Arbeitsaufwand beträgt insgesamt 360 Std., welche sich grob wie folgt gliedern: <ul style="list-style-type: none"> • 60 Std. Recherche, Planung und Teilnahme am Planungworkshop • 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien • 180 Std. Durchführung des Projekts (Projektarbeit) • 20 Std. Erstellung des Zwischenberichts (Hausarbeit) • 60 Std. Erstellung des Abschlussberichts, sowie Erstellung und Präsentation des Projektposters (Hausarbeit und Kolloquium) 	
Empfohlene Vorkenntnisse: Grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten und zum wissenschaftlichen Arbeiten, sowie Grundkenntnisse in der Programmierung paralleler und verteilter Systeme wie sie z.B. durch DSG-PKS-B und/oder DSG-IDistrSys vermittelt werden.	Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung

Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 2 Semester
---	----------------------------------	---

Lehrveranstaltungen	
<p>Übung DSG Bachelorprojekt Software Systems Science Lehrformen: Übung Dozenten: Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik Sprache: Deutsch/Englisch Angebotshäufigkeit: WS, SS</p> <hr/> <p>Lernziele: siehe Modulbeschreibung</p> <hr/> <p>Inhalte: vgl. Modulbeschreibung</p> <hr/> <p>Literatur: Je nach Projektthematik; wird zu Beginn des Projekts bekannt gegeben.</p>	8,00 SWS

<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 6 Monate</p> <p>Beschreibung: Kolloquium: Fachliche Diskussion auf der Grundlage des im Projekt bearbeiteten Themas im Rahmen einer Abschlussveranstaltung, auf der zunächst das zum Projekt angefertigte Poster erläutert wird; hier können auch praktische Projektergebnisse (z. B. lauffähige Software) demonstriert werden. Hausarbeit: Anfertigen eines schriftlichen Abschlussberichts nach abgeschlossener Projektarbeit.</p>	
--	--

<p>Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 6 Monate</p> <p>Beschreibung: Anfertigen eines schriftlichen Zwischenberichts zum Projekt nach etwa 80 Std. geleisteter Projektarbeit, spätestens am Ende des Semesters, in dem das Projekt begonnen wurde.</p>	
---	--

Modul EESYS-IITP-B Internationales IT-Projektmanagement <i>International IT Project Management</i>		6 ECTS / 180 h
Version 1.0.0 (seit SS13) Modulverantwortliche/r: Prof. Dr. Thorsten Staake, Anna Kupfer		
Inhalte: Die Studierenden erhalten ein Verständnis über die grundlegenden Methoden des IT-Projektmanagements. Besonderheiten internationaler Projekte werden ebenfalls diskutiert.		
Lernziele/Kompetenzen: Studierende sollen nach erfolgreichem Abschluss der Veranstaltung in der Lage sein, IT-Projekte in kleinen und großen Organisationen zu initiieren, planen, leiten und zu überwachen.		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Internationales IT-Projektmanagement Lehrformen: Vorlesung Dozenten: Prof. Dr. Thorsten Staake Sprache: Deutsch Inhalte: Gegenstand der Lehrveranstaltung sind Grundlagen des Managements von IT-Projekten mit ausgewählten Vertiefungen zu den spezifischen Aspekten internationaler Projekte. Eingenommen wird sowohl die Perspektive kleiner Unternehmen/Startups und großer Organisationseinheiten mit etablierten Prozessen. Die Lehrveranstaltung geht auf die Initiierung, Planung, Durchführung und das Controlling von IT-Projekten ein.	2,00 SWS
2. Übung Internationales IT-Projektmanagement Lehrformen: Übung Dozenten: Anna Kupfer Sprache: Deutsch Inhalte: Anwendungen und Vertiefungen zu den Inhalten der Vorlesung, teilweise in Kleingruppen; wenn es die Anzahl der Teilnehmerinnen und Teilnehmer gestattet, wird ein konkretes Projekt bearbeitet.	2,00 SWS

Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten Beschreibung:	
--	--

In der Klausur werden die in Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. Die Bearbeitungsdauer der Klausur beträgt 90 Minuten. Durch die freiwillige Bearbeitung von semesterbegleitenden Studienleistungen können Teilnehmende 12 Punkte sammeln, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Mögliche Studienleistungen sind schriftliche Hausarbeiten, Referate oder kleinere Software-Projekte. Zu Beginn der Lehrveranstaltung werden die Bearbeitungsdauer und die Punkte pro optionaler Studienleistung angegeben. Eine Bewertung von 1.0 kann auch ohne Punkte aus den Übungen erreicht werden.

Modul Gdl-GTI-B Grundlagen der Theoretischen Informatik		6 ECTS / 180 h
<i>Machines and Languages</i>		
Version 1.0.0 Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Inhalte: In der Veranstaltung wird die Theorie der Automaten, Sprachen und Algorithmen in ihren Grundzügen entwickelt. Das intuitiv einfach zu erfassende Modell der Turingmaschine als das Standardmodell der Berechenbarkeit und historischer Ausgangspunkt für die Entwicklung von programmierbaren Rechenmaschinen sowie der Lambda-Kalkül als Basis zum Verständnis funktionaler und anderer höherer Programmiersprachen stehen dabei im Mittelpunkt. Mit Turingmaschinen und anderer damit äquivalenter Berechnungsmodelle stößt die Veranstaltung zur Grenze dessen vor, was nach heutigem Wissen als prinzipiell maschinell berechenbar angesehen wird. Hierbei werden die wichtigsten Begriffe der Berechenbarkeits- und Komplexitätstheorie, etwa die Chomsky Hierarchie und die P/NP Komplexitätsklassen, besprochen. Über die klassischen Modelle der Algorithmentheorie hinaus werden, je nach verfügbarer Zeit, auch neuere Semantiken für nebenläufige und verteilte sowie objektorientierte Programmierung eingeführt und an Beispielen diskutiert.		
Lernziele/Kompetenzen: Kenntnis der wichtigsten Ergebnisse der Berechenbarkeits- und Komplexitätstheorie und den damit zusammenhängenden grundlegenden Einsichten in die Struktur und die Grenzen der Berechenbarkeit; Fähigkeit, Berechnungsmodelle unterschiedlicher Ausdruckskraft systematisch aufeinander zu reduzieren und die Turing-Äquivalenz von Programmiersprachen nachzuweisen oder zu widerlegen; Kenntnis konkreter mathematischer Grundmodelle zur Beschreibung von Algorithmus und Prozess, welche die wissenschaftlich-methodische Basis der Informatik bilden; Fähigkeit, rekursive und iterative Problemlösungen einerseits, sowie funktionale und reaktive Vorgänge andererseits gegeneinander abzugrenzen und ihre jeweilige Angemessenheit für die Modellierung praktischer Steuerungs- und Datenverarbeitungsaufgaben zu erkennen.		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: 45 Stunden • Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden • Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden • Prüfungsvorbereitung: 30 Stunden • schriftliche Prüfung: 90 Minuten 		
Empfohlene Vorkenntnisse: gute Englischkenntnisse Modul Einführung in die Informatik (DSG-Eidl-B) - empfohlen Modul Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
<p>1. Vorlesung Grundlagen der Theoretischen Informatik</p> <p>Lehrformen: Vorlesung</p> <p>Dozenten: Prof. Ph.D. Michael Mendler</p> <p>Sprache: Deutsch/Englisch</p>	2,00 SWS
<p>Inhalte:</p> <p>In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.</p>	
<p>Literatur:</p> <ul style="list-style-type: none"> • Hopcroft, J. E., Motwani, R., Ullman, J. D.: Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, Pearson Studium, 2002. • Asteroth, A., Baier, Ch.: Theoretische Informatik, Pearson Studium, 2002. • Martin, J. C.: Introduction to Languages and the Theory of Computation, McGraw Hill, (2nd ed.), 1997. 	
<p>2. Übung Grundlagen der Theoretischen Informatik</p> <p>Lehrformen: Übung</p> <p>Dozenten: Prof. Ph.D. Michael Mendler, N.N.</p> <p>Sprache: Englisch/Deutsch</p>	2,00 SWS
<p>Inhalte:</p> <p>Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p>	
<p>Prüfung</p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

Modul Gdl-Mfl-1 Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) <i>Propositional and Predicate Logic</i>		6 ECTS / 180 h
Version 1.0.0 Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Inhalte: In dieser Basisvorlesung werden die für die Informatik wesentlichen Elemente der Aussagen- und Prädikatenlogik, sowie ihre Anwendung zur Spezifikation und Analyse diskreter Strukturen eingeführt. Am Beispiel der Prädikatenlogik wird der Prozess der Abstraktion im Aufbau und der Anwendung von formalen Systemen eingehend dargestellt. Der zentrale Unterschied zwischen Syntax und Semantik und das Prinzip rekursiver Konstruktionen und induktiven Schließens werden dabei ausführlich erläutert.		
Lernziele/Kompetenzen: Die Fähigkeit, informell gegebene Strukturen und Prozesse der natürlichen und technischen Umwelt, speziell solche mit nicht-numerischem Charakter mit symbolischen Formalismen zu erfassen und mit Hilfe kombinatorischer und logischer Lösungsansätze zu analysieren; Die Fähigkeit zur Abstraktion und die Einsicht in die methodische Bedeutung des hierarchischen Aufbaus informatischer Systeme, des systematischen Fortschreitens von einfachen zu komplexen Beschreibungen sowie umgekehrt des inkrementellen Abstützens komplexer Problemlösungen auf elementare Lösungsbausteine; Die Kenntnis elementarer Grundbegriffe der Beweis- und Modelltheorie der klassischen Aussagen- und Prädikatenlogik.		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: 45 Stunden • Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 60 Stunden • Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 45 Stunden • Prüfungsvorbereitung: 30 Stunden • schriftliche Prüfung: 90 Minuten 		
Empfohlene Vorkenntnisse: gute Englischkenntnisse		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Mathematik für Informatiker 1 Lehrformen: Vorlesung Dozenten: Prof. Ph.D. Michael Mendler Sprache: Deutsch/Englisch	2,00 SWS
Inhalte: In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.	

<p>Literatur:</p> <ul style="list-style-type: none"> • Ehrig, H., Mahr, B., Cornelius, F., Große-Rhode, Zeitz, M. P.: Mathematisch strukturelle Grundlagen der Informatik. Springer Verlag, 2. Aufl., 2001. • Grassmann, W. K., Tremblay, J.-P.: Logic and Discrete Mathematics - A Computer Science Perspective. Prentice Hall, 1996. • Scheinerman, E. R.: Mathematics – A Discrete Introduction. Brooks/Cole, 2000. • Barwise, J., Etchemendy, J: Language, Proof, and Logic. Seven Bridges Press, 2000. 	
<p>2. Übung Mathematik für Informatiker 1 Lehrformen: Übung Dozenten: Prof. Ph.D. Michael Mendler, N.N. Sprache: Deutsch/Englisch</p>	2,00 SWS
<p>Inhalte: Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p>	
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

Modul GdI-NPP-B Nichtprozedurale Programmierung <i>Functional Programming</i>		6 ECTS / 180 h
Version 1.0.0 Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Inhalte: Die Veranstaltung beschäftigt sich mit den Grundlagen der logischen und funktionalen Programmierung als die wichtigsten Alternativen zu herkömmlichen prozeduralen Sprachen. Diese nichtprozeduralen Sprachen, welche dem deklarativen und rekursiven Programmierprinzip folgen, werden besonders für ihre hohe Programmiereffizienz und -Sicherheit geschätzt. Der systematische Aufbau einer funktionalen Programmiersprache wird schrittweise erläutert und anhand konkreter Aufgabenstellungen nachvollzogen. Ausführliche praktische Übungen mit der Programmiersprache Haskell ergänzen die theoretischen Inhalte. Besonderes Augenmerk wird auf die Einführung in polymorphe Typsysteme gelegt und ihre Anwendung in der Typprüfung und Typsynthese als automatisches Softwarevalidierungsverfahren. An Beispielen wird die deklarative Programmierung interaktiver Anwendungen nach dem synchronen Programmierprinzip (synchrone Kahn-Netzwerke) aufgezeigt.		
Lernziele/Kompetenzen: Fähigkeit zur Entwicklung algorithmischer Problemlösungen in nichtprozeduralen Programmiersprachen; Einsicht in die Bedeutung formaler Semantiken für die Implementierung von Programmiersprachen und die Fähigkeit, die funktionale Korrektheit einfacher Programme über ihre formale Semantik zu verifizieren; Kenntnis verschiedener Techniken zur Semantikgebung, insbesondere die denotationelle, operationelle, und Termersetzungsemantik; die Fähigkeit neue Sprachkonstrukte mit diesen Techniken zu spezifizieren; Fähigkeit, sich neue Programmiersprachen systematisch zu erarbeiten und diese in ihren Anwendungsmöglichkeiten kompetent einzuordnen; Kenntnis deklarativer Modelle interaktiver Software und die Fähigkeit, diese in einer konkreten Programmiersprache zu implementieren.		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: 45 Stunden • Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 60 Stunden • Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 45 Stunden • Prüfungsvorbereitung: 30 Stunden • schriftliche Prüfung: 90 Minuten 		
Empfohlene Vorkenntnisse: gundlegende Programmierkenntnisse, gute Englischkenntnisse Modul Einführung in die Informatik (DSG-EidI-B) - empfohlen Modul Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (GdI-MfI-1) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen

1. Vorlesung Nichtprozedurale Programmierung

<p>Lehrformen: Vorlesung Dozenten: Prof. Ph.D. Michael Mendler Sprache: Englisch/Deutsch</p>	<p>2,00 SWS</p>
<p>Inhalte: In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.</p>	
<p>Literatur:</p> <ul style="list-style-type: none"> • Pierce, B. C.: Types and Programming Languages, MIT Press, 2002 • Thompson, S.: Haskell – The Craft of Functional Programming, Addison-Wesley 1999. • O’Keefe, R. A.: The Craft of Prolog. MIT Press, 2nd printing, 1994. 	
<p>2. Übung Nichtprozedurale Programmierung Lehrformen: Übung Dozenten: Prof. Ph.D. Michael Mendler Sprache: Englisch/Deutsch</p>	<p>2,00 SWS</p>
<p>Inhalte: Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p>	
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

Modul Gdl-PR2-B Bachelorprojekt Software Systems Science		12 ECTS / 360 h
<i>Bachelor Project Software Systems Science</i>		
Version 1.0.0 Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Inhalte: Ein überschaubares Projekt zu Themen aus dem Bereich der aktuellen Forschungsarbeiten der Professur wird in arbeitsteilig arbeitenden Kleingruppen von Studierenden, ggf. auch einzeln - von der Konzeption und Umsetzung bis zur Dokumentation in einem wissenschaftlichen Arbeitsbericht und einer Präsentation - durchgeführt. Dies erfordert eine umfangreiche Recherche der aktuellen wissenschaftlichen Literatur sowie eine detaillierte Planung, deren Ergebnisse in einem Planungsworkshop vorgestellt und diskutiert werden.		
Lernziele/Kompetenzen: Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Softwaresystem-Projekten auftretenden konzeptionellen und praktischen Problem wie auch von geeigneten Lösungsansätzen erhalten. Dies geschieht in der intensiven Bearbeitung einer Aufgabenstellung mit Bezug zu den Grundlagen der Informatik in Kleingruppen - ggf. auch einzeln. Die Studenten sammeln wichtige Erfahrungen in der Durchführung kleinerer forschungsorientierter Projekte.		
Bemerkung: Dieses Modul soll über einen Zeitraum von 2 Semestern bearbeitet werden (2x6ECTS, 2x4SWS). In besonderen Fällen, z.B. Abwesenheit aufgrund eines Auslandsstudiums, kann das Projekt auch innerhalb eines Semesters plus Semesterferien durchgeführt werden. Studenten werden gebeten in solchen Fällen das Thema und den zeitlichen Ablauf schon im Semester vor Beginn des Projektes mit dem betreuenden Dozenten festzulegen. Der Arbeitsaufwand beträgt insgesamt 360 Std, welche sich grob wie folgt aufteilen: <ul style="list-style-type: none"> • 60 Std. Recherche, Planung und Teilnahme am Planungsworkshop • 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien • 180 Std. Durchführung des Projekts • 20 Std. Erstellung des Zwischenberichts (Hausarbeit) • 60 Std. Erstellung Abschlussbericht, sowie Erstellung und Präsentation des Projektposters (Hausarbeit und Kolloquium) Berichte und Präsentation dürfen wahlweise in Deutsch oder Englisch abgefasst sein.		
Empfohlene Vorkenntnisse: Grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z.B. erworben im Modul "Software Engineering Lab" und zum wissenschaftlichen Arbeiten, z.B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik". Darüber hinaus sind für Projekte im Bereich Gdl empfohlen: Englischkenntnisse, Kenntnisse der elementaren Logik, Grundlagen der Theoretischen Informatik, Rechner- und Betriebssysteme, Nichtprozedurale Programmierung.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester:	Minimale Dauer des Moduls:

		2 Semester
Lehrveranstaltungen		
Übung Gdl Bachelorprojekt Software Systems Science Lehrformen: Übung Dozenten: Prof. Ph.D. Michael Mandler, N.N. Sprache: Deutsch/Englisch		8,00 SWS
Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 6 Monate Beschreibung: Anfertigen eines schriftlichen Abschlussberichts und Kolloquium (Fachgespräch) zu den Ergebnisse des Projektes. Die Gewichtung der Prüfungsleistungen Hausarbeit und Kolloquium wird zu Beginn der Lehrveranstaltung von der Dozentin bzw. dem Dozenten bekanntgegeben		
Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 4 Monate Beschreibung: Anfertigen eines schriftlichen Zwischenberichts zum Projekt nach etwa 80 Std. geleisteter Projektarbeit, spätestens am Ende des Semesters, in dem das Projekt begonnen wurde.		

Modul Gdl-SaV-B Logik (Specification and Verification)		6 ECTS / 180 h
<i>Specification and Verification</i>		
Version 1.0.0 (seit WS14/15 bis SS15) Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Inhalte: Nicht nur die Verifikation der funktionalen Korrektheit von Algorithmen und die funktionale Analyse verteilter und verlässlicher Systeme erfordert logisch-symbolische Verfahren. Auch viele Steuerungsprobleme in Anwendungsfeldern wie der Automatisierung von Wirtschaftsprozessen, intelligenten autonomen Agenten oder in Sicherheitsprotokollen lassen sich nur schwer mit herkömmlichen analytisch-numerischen Methoden behandeln. Dank der sich kontinuierlich verbessernden Leistungsfähigkeit moderner Rechner und der Erfolge im Gebiet der <i>Computational Logic</i> kommt der formalen Logik in der Informationstechnik wachsende Bedeutung zu. Die Vorlesung gibt einen Einblick in die Familie der Modallogiken als die wichtigsten informatikrelevanten Logiken, stellt zugehörige Implementierungstechniken und Entscheidungsverfahren vor und zeigt typische Anwendungen auf.		
Lernziele/Kompetenzen: Einsicht in die besondere Stellung der Modallogik zwischen Aussagenlogik und Prädikatenlogik und die Kenntnis ihrer ingenieurtechnischen Einsatzmöglichkeiten in Anwendungen, etwa der semantischen Informationsverarbeitung oder der Verifikation robuster und funktionssicherer reaktiver Systeme; Kenntnis der wichtigsten Modallogiken, ihrer Ausdruckskraft und Automatisierbarkeit, sowie die Fähigkeit für vorgegebene Anwendungen maßgeschneiderte Modallogiken selbst zu entwickeln; Fähigkeit, dynamische und reaktive Abläufe sowie komplexe verteilte Kommunikationsvorgänge in modaler und temporaler Logik zu spezifizieren und diese mit Hilfe geeigneter formaler Kalküle zu analysieren.		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: 45 Stunden • Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden • Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden • Prüfungsvorbereitung: 30 Stunden 		
Empfohlene Vorkenntnisse: grundlegende Programmierkenntnisse, gute Englischkenntnisse Modul Einführung in die Informatik (DSG-Eidl-B) - empfohlen Modul Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
Lehrveranstaltungen		
Vorlesung und Übung Logik (Specification and Verification) Lehrformen: Vorlesung und Übung Dozenten: Prof. Ph.D. Michael Mendler Sprache: Englisch/Deutsch		4,00 SWS

<p>Inhalte: In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben. Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p> <hr/> <p>Literatur:</p> <ul style="list-style-type: none">• Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y.: Reasoning about Knowledge. MIT Press, (2nd printing) 1996.• Hughes, G. E., Cresswell, M. J.: A New Introduction to Modal Logic. Routledge, (3rd reprint) 2003.• Popkorn, S.: First Steps in Modal Logic. Cambridge University Press, 1994.• Berard, B., Bidoit, M., Finkel, A., Laroussinie, F., Petit, A., Petrucci, L., Schnoebelen, Ph., McKenzie, P.: Systems and Software Verification. Springer 1999.	
---	--

<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	
--	--

Modul HCI-DISTP-B Design Interaktiver Systeme: Theorie und Praxis <i>Design of Interactive Systems: Theory and Practice</i>		3 ECTS / 90 h
Version 1.0.0 (seit SS13) Modulverantwortliche/r: Prof. Dr. Tom Gross		
Inhalte: Theoretische, methodische und praktische Grundlagen des Entwurfs sowie praktisches Entwerfen einer forschungsrelevanten Aufgabenstellung.		
Lernziele/Kompetenzen: Ziel ist die allgemeine Einführung und Vermittlung grundlegender Praktiken, Prozesse und Methoden des Designs mit besonderem, anwendungsbezogenem Fokus auf die nutzerzentrierte Gestaltung komplexer, interaktiver Systeme		
Bemerkung: http://www.uni-bamberg.de/hci/leistungen/studium Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesungseinheiten • Teilnahme an Gruppenbesprechungen • Bearbeitung der Projektaufgabenstellung allein und im Team • Vorbereitung von Besprechungen und Präsentationen • Prüfungsvorbereitung Die Aufwände können dabei in Abhängigkeit von der Aufgabenstellung und der in der Gruppe abgestimmten Aufgabenverteilung unter den Gruppenmitgliedern unterschiedlich auf die Bereiche verteilt sein.		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Vorlesung Design Interaktiver Systeme: Theorie und Praxis Lehrformen: Vorlesung Dozenten: Jochen Denzinger Sprache: Deutsch	1,00 SWS
Inhalte: Im Rahmen der Vorlesung werden folgende Themen behandelt: <ul style="list-style-type: none"> • Designtheorie und -geschichte • Gestaltung multimodaler Benutzungsoberflächen • User-Centered Design, User Experience Design • Entwurfspraxis inkl. praktischer Einsatz von Methoden für den iterativen Entwurf 	

<p>Im Praktikum werden wechselnde Projekte zu den Inhalten der Lehrveranstaltung bearbeitet. Im Rahmen der Veranstaltung ist ein iterativer Entwurf als praktische Übung von den Studierenden zu erstellen. Die in einem Projektpraktikum bearbeitete Aufgabenstellung geht deutlich über den Umfang einer normalen Übungsaufgabe hinaus und wird in kleinen Gruppen bearbeitet. Das erarbeitete Ergebnis wird dokumentiert und in einer Abschlusspräsentation vorgestellt.</p>	
<p>Literatur: Die Veranstaltung ist eine Zusammensetzung verschiedener Quellen; als ergänzende Quellen und zum Nachschlagen wir u.a. empfohlen:</p> <ul style="list-style-type: none">• Krippendorff, K. The Semantic Turn. A New Foundation for Design. Taylor & Francis Group, Boca Raton, FL, 2006.• Moggridge, B. Designing Interactions. MIT Press, Cambridge, MA, 2007.	

<p>Prüfung Kolloquium, Design interaktiver Systeme: Theorie und Praxis / Prüfungsdauer: 30 Minuten</p> <p>Beschreibung: Kolloquium zum Projektverlauf und Projektergebnissen</p>	
--	--

Modul HCI-IS-B Interaktive Systeme <i>Interactive Systems</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS11/12) Modulverantwortliche/r: Prof. Dr. Tom Gross		
Inhalte: Theoretische, methodische und praktische Grundlagen der Mensch-Computer-Interaktion.		
Lernziele/Kompetenzen: Ziel ist die allgemeine Einführung und Vermittlung von grundlegenden Paradigmen, Konzepten und Prinzipien der Gestaltung von Benutzungsoberflächen. Der primäre Fokus liegt dabei auf dem Entwurf, der Implementation und der Evaluierung von interaktiven Systemen.		
Bemerkung: http://www.uni-bamberg.de/hci/leistungen/studium Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden • Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden • Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden • Bearbeiten der optionalen Studienleistungen: insgesamt ca. 45 Stunden • Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) 		
Empfohlene Vorkenntnisse: Grundkenntnisse in Informatik im Umfang einer Einführung in die Informatik		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Interaktive Systeme Lehrformen: Vorlesung Dozenten: Prof. Dr. Tom Gross Sprache: Deutsch	2,00 SWS
Inhalte: Im Rahmen der Vorlesung werden die folgenden Themen behandelt: <ul style="list-style-type: none"> • Einführung in die Gestaltung von Benutzungsoberflächen • Benutzer und Humanfaktoren • Maschinen und technische Faktoren • Interaktion, Entwurf, Prototyping und Entwicklung • Evaluierung von interaktiven Systemen • Entwicklungsprozess interaktiver Systeme • Interaktive Systeme im größeren Kontext und verwandte Themen 	
Literatur:	

<p>Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> • Preece, J., Rogers, Y. und Sharp, H. Interaction Design: Beyond Human-Computer Interaction. Wiley, New York, NY, 3. Auflage, 2011 • Dix, A., Finlay, J., Abowd, G.D. und Beale, R. Human-Computer Interaction. Pearson, Englewood Cliffs, NJ, 3. Auflage, 2004. 	
<p>2. Übung Interaktive Systeme Lehrformen: Übung Dozenten: Mitarbeiter Mensch-Computer-Interaktion Sprache: Deutsch</p> <hr/> <p>Inhalte: praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <hr/> <p>Literatur: siehe Vorlesung</p>	<p>2,00 SWS</p>
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: In der Klausur können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

Modul HCI-KS-B Kooperative Systeme <i>Cooperative Systems</i>		6 ECTS / 180 h
Version 1.0.0 (seit SS11) Modulverantwortliche/r: Prof. Dr. Tom Gross		
Inhalte: Theoretische, methodische und praktische Grundlagen der rechnergestützten Gruppenarbeit.		
Lernziele/Kompetenzen: Ziel ist die Vermittlung von grundlegenden Paradigmen und Konzepten von Rechnergestützter Gruppenarbeit (Computer-Supported Cooperative Work; CSCW) sowie die daraus resultierenden Designprinzipien und Prototypen. Dabei wird der Begriff breit gefasst; das zentrale Anliegen ist entsprechend die generelle technische Unterstützung von sozialer Interaktion, welche vom gemeinsamen Arbeiten und Lernen bis zum privaten Austausch reichen kann.		
Bemerkung: http://www.uni-bamberg.de/hci/leistungen/studium Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden • Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden • Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen: ca. 30 Stunden • Bearbeiten der optionalen Studienleistung: insgesamt ca. 45 Stunden • Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) 		
Empfohlene Vorkenntnisse: Grundkenntnisse in Informatik im Umfang einer Einführung in die Informatik sowie Programmierkenntnisse in Java.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Kooperative Systeme Lehrformen: Vorlesung Dozenten: Prof. Dr. Tom Gross Sprache: Deutsch	2,00 SWS
Inhalte: Im Rahmen der Vorlesung werden nach einer Einführung in das Thema die folgenden Themen behandelt: <ul style="list-style-type: none"> • Grundlegende Konzepte • Technologische Unterstützung für wechselseitige Information, Kommunikation, Koordination, Gruppenarbeit und Online-Gemeinschaften • Analyse kooperativer Umgebungen • Entwurf von CSCW und Groupware • Implementation von CSCW und Groupware 	

<ul style="list-style-type: none"> • CSCW im größeren Kontext und verwandte Themen <p>Literatur: Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> • Gross, T. und Koch, M. Computer-Supported Cooperative Work. Oldenbourg, München, 2007. • Borghoff, U.M. und Schlichter, J.H. Computer-Supported Cooperative Work: Introduction to Distributed Applications. Springer-Verlag, Heidelberg, 2000. 	
<p>2. Übung Kooperative Systeme</p> <p>Lehrformen: Übung</p> <p>Dozenten: Mitarbeiter Mensch-Computer-Interaktion</p> <p>Sprache: Deutsch</p> <p>Inhalte: praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <p>Literatur: siehe Vorlesung</p>	2,00 SWS
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: In der Klausur können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

Modul HCI-US Ubiquitäre Systeme <i>Ubiquitous Systems</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS11/12) Modulverantwortliche/r: Prof. Dr. Tom Gross		
Inhalte: Theoretische, methodische und praktische Grundlagen des Ubiquitous Computing.		
Lernziele/Kompetenzen: Ziel ist die Vermittlung fundierter Kenntnisse und Fähigkeiten im Bereich der ubiquitären Systeme sowie eines breiten theoretischen und praktischen Methodenwissens zum Entwurf, zur Konzeption und zur Evaluierung ubiquitärer Systeme. Nach dem Besuch dieser Lehrveranstaltung sollen Studierende die einschlägige Literatur und Systeme in Breite und Tiefe kennen und neue Literatur und Systeme kritisch bewerten können.		
Bemerkung: http://www.uni-bamberg.de/hci/leistungen/studium Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden • Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden • Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen sowie Bearbeitung der optionalen Studienleistungen): ca. 75 Stunden • Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) 		
Empfohlene Vorkenntnisse: keine Modul Einführung in die Informatik (DSG-Eidl-B) - empfohlen Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Ubiquitäre Systeme Lehrformen: Vorlesung Dozenten: Prof. Dr. Tom Gross Sprache: Deutsch	2,00 SWS
Inhalte: Im Rahmen der Vorlesung werden nach einer Einführung in das Thema Ubiquitous Computing - also der allgegenwärtigen Rechner, die verschwindend klein, teilweise in Alltagsgegenständen eingebaut, als Client und Server fungieren und miteinander kommunizieren können - die folgenden Themen konzeptionell, technisch und methodisch behandelt: <ul style="list-style-type: none"> • Grundlegende Konzepte • Basistechnologie und Infrastrukturen • Ubiquitäre Systeme und Prototypen 	

<ul style="list-style-type: none"> • Kontextadaptivität • Benutzerinteraktion • Ubiquitäre Systeme im größeren Kontext und verwandte Themen <p>Literatur: Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> • Krumm, J., (Hrsg.). Ubiquitous Computing Fundamentals. Taylor & Francis Group, Boca Raton, FL, 2010. 	
<p>2. Übung Ubiquitäre Systeme</p> <p>Lehrformen: Übung</p> <p>Dozenten: Mitarbeiter Mensch-Computer-Interaktion</p> <p>Sprache: Deutsch</p> <p>Inhalte: praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <p>Literatur: siehe Vorlesung</p>	2,00 SWS
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: In der Klausur können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

Modul IAI-WAI-B Wissenschaftliches Arbeiten in der Informatik <i>Scientific Methods in Computer Science</i>		3 ECTS / 90 h
Version 2.0.0 (seit SS14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen		
Inhalte: Dieses Modul richtet sich an Studierende der Studiengänge B.Sc. Angewandte Informatik und B.Sc. Software Systems Science sowie interessierte Studierende anderer Studiengänge, die im Bereich Informatik eine Projekt-, Seminar-, Bachelor- oder Masterarbeit schreiben möchten. Das Modul führt diese Studierenden in zentrale Methoden, Techniken und Werkzeuge des wissenschaftlichen Arbeitens theoretisch und praktisch ein.		
Lernziele/Kompetenzen: Nach erfolgreicher Teilnahme an diesem Modul können Studierende insbesondere: <ul style="list-style-type: none"> • wissenschaftliches Arbeiten planen und managen; • Literaturrecherchen selbständig durchführen und die Güte verschiedener Quellen einschätzen; • den Aufbau einer wissenschaftlichen Arbeit anhand formaler Anforderungen erkennen und beurteilen; • die Gliederung, die Problemstellung und das Literaturverzeichnis einer wissenschaftlichen Arbeit erstellen; • typische Forschungsmethoden der Informatik einordnen; • wissenschaftliche Vorträge vorbereiten und halten; • elementare Softwarewerkzeuge zur Unterstützung des wissenschaftlichen Arbeitens einsetzen. 		
Bemerkung: Der Arbeitsaufwand beträgt 90 Std., welche sich grob wie folgt gliedern: <ul style="list-style-type: none"> • 30 Std. Teilnahme an der Vorlesung und Übung • 30 Std. Vor- und Nachbereitung der Vorlesung und Übung • 30 Std. Erstellung der schriftlichen Hausarbeit 		
Empfohlene Vorkenntnisse: Keine. Der Besuch des Moduls im 1. Fachsemester wird allerdings nicht empfohlen.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Vorlesung/Übung Wissenschaftliches Arbeiten in der Informatik Lehrformen: Vorlesung und Übung Dozenten: Prof. Dr. Gerald Lüttgen Sprache: Deutsch	2,00 SWS
Inhalte: Die Vorlesung/Übung bietet den Studierenden einen Einblick in zentrale Themenbereiche des wissenschaftlichen Arbeitens: <ul style="list-style-type: none"> • Wissenschaftsethik und wissenschaftliche Qualitätskriterien 	

<ul style="list-style-type: none">• Wissenschaftliche Arbeiten: Arten, Aufbau und Bewertung• Literatur: Recherchieren, lesen und richtig zitieren• Wissenschaftliches Schreiben• Forschungsmethoden der Informatik• Projektmanagement am Beispiel "Abschlussarbeit": von der Themenfestlegung bis zur Abgabe• Wissenschaftliche Vorträge vorbereiten und halten <p>Begleitend werden diese theoretischen Inhalte anhand eines konkreten wissenschaftlichen Themas praktisch eingeübt und vertieft. Dabei werden auch verschiedene Software-Werkzeuge – z. B. zur Literaturverwaltung – vorgestellt.</p> <p>Literatur:</p> <ul style="list-style-type: none">• Balzert, H., Schröder, M. und Schäfer, C. Wissenschaftliches Arbeiten, 2. Auflage, Herdecke/Witten, W3L, 2011.• Franck, N. und Stary, J. Die Technik wissenschaftlichen Arbeitens, 16. Auflage, Paderborn, Schöningh, 2011.	
--	--

<p>Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 3 Wochen</p> <p>Beschreibung: Selbständiges Anwenden der in der Vorlesung und Übung vermittelten Inhalte auf eine kleine Fallstudie, deren Ausgangspunkt beispielsweise eine simple Forschungsfrage ist.</p>	
--	--

Modul KInf-GeoInf-B Geoinformationssysteme		6 ECTS / 180 h
Version 1.0.0 (seit WS12/13) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder		
Inhalte: Das Modul führt ein in die Grundlagen der Geoinformationsverarbeitung. Es besteht aus zwei Lehrveranstaltungen: einer Vorlesung, die Konzepte und Methoden vermittelt sowie einer Übung, in der die Anwendung der Methoden auf konkrete Problemstellungen eingeübt wird. Eine weitergehende Inhaltsbeschreibung findet sich bei den Lehrveranstaltungen.		
Lernziele/Kompetenzen: Die Studierenden lernen Grundbegriffe sowie wichtige Methoden aus dem Bereich der Geoinformationssysteme kennen. Sie erwerben folgende Kompetenzen: <ul style="list-style-type: none"> • fachliche Anforderungen im Hinblick auf die Geodatenmodellierung zu analysieren und passende Geodatenmodelle zu erstellen • geoinformatische Analyseverfahren vergleichend zu bewerten und die für ein Anwendungsproblem geeigneten Verfahren zu identifizieren. 		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Vorlesung und Übung: 45 Stunden • Vor- und Nachbereitung der Vorlesung: 30 Stunden • Vor- und Nachbereitung der Übung inkl. Recherche und Studium zusätzlicher Quellen aber ohne Bearbeitung der Übungsaufgaben: 30 Stunden • Bearbeiten der Übungsaufgaben: 45 Stunden • Prüfungsvorbereitung: 30 Stunden 		
Empfohlene Vorkenntnisse: Grundkenntnisse der Informatik, wie sie in den empfohlenen Modulen vermittelt werden Modul Einführung in die Informatik (DSG-Eidl-B) - empfohlen Modul Informatik und Programmierkurs für die Kulturwissenschaften (KInf-IPKult-E) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Geoinformationssysteme Lehrformen: Vorlesung Dozenten: Prof. Dr. Christoph Schlieder Sprache: Deutsch	2,00 SWS
Inhalte: Geoinformationssysteme (GIS) dienen der effizienten Erfassung, Analyse und Bereitstellung georeferenzierter Daten. Die Lehrveranstaltung stellt die grundlegenden Konzepte vor, die der Modellierung von Geodaten zugrunde	

<p>liegen. Hierzu gehört z.B. die unterschiedliche Repräsentation räumlicher Objekte in Vektor- und Raster-GIS. Weitere Themen sind die Geodaten-Erfassung sowie Ansätze zur Geodatensvisualisierung. Anwendungen der Geoinformationsverarbeitung werden an klassischen Einsatzfeldern (Umweltinformationssysteme) und aktuellen technologischen Entwicklungen (mobile Computing) illustriert. Querverbindungen zum Bereich der Semantischen Informationsverarbeitung ergeben sich vor allem im Zusammenhang mit der Interoperabilität von GIS.</p>	
<p>Literatur: Longley, P., Goodchild, M., Maguire, D., Rhind, D. (2001): Geographic Information: Systems and Science, Wiley: Chichester, UK. Shekhar, S., Chawla, S. (2003): Spatial Databases: A Tour, Prentice Hall: Upper Saddle River, NJ. Smith, M., Goodchild, M., and Longley, P. (2007): Geospatial Analysis, 2nd edition, Troubador Publishing Ltd.</p>	
<p>2. Übung Geoinformationssysteme Lehrformen: Übung Dozenten: Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften Sprache: Deutsch</p> <hr/> <p>Inhalte: siehe Vorlesung</p> <hr/> <p>Literatur: siehe Vorlesung</p>	<p>2,00 SWS</p>
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten Beschreibung: In der schriftlichen Prüfung werden die in Vorlesung und Übung behandelten Themengebiete geprüft.</p>	

Modul KInf-SemInf-M Semantic Information Processing <i>Semantic Information Processing</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS12/13) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder		
Inhalte: The module introduces students into the research field of semantic information processing. It consists of two parts, a lecture (Vorlesung) which covers the basic methods and lab sessions in which the methods are applied to problems (Übung). For more detail refer to the content description of the lecture.		
Lernziele/Kompetenzen: After completion of this module, students will be able to <ul style="list-style-type: none"> • explain and compare the fundamental concepts of semantic information processing • describe and analyze methods for problem solving by heuristic search • critically discuss different approaches to knowledge representation • select algorithms that are appropriate for a given type of application problem 		
Bemerkung: The main language of instruction in this course is English. The lab sessions may be delivered in German if all participating students are fluent in German. The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 45 hrs. attending lecture and lab sessions • 30 hrs. preparing and reviewing the lectures • 30 hrs. preparing and reviewing the lab sessions • 45 hrs. working on the written assignment • 30 hrs. preparation for the exam 		
Empfohlene Vorkenntnisse: Students are expected to come with general programming skills and to be familiar with formal methods in computer science.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Lectures on Semantic Information Processing Lehrformen: Vorlesung Dozenten: Prof. Dr. Christoph Schlieder Sprache: Englisch/Deutsch	2,00 SWS
Inhalte: Semantic information processing addresses problems in which software systems need to represent knowledge, not just data. Facts from different knowledge sources are combined and integrated by machine reasoning processes. The services of the Semantic Web provide a prominent example for applications that make extensive use of knowledge representation and reasoning. The lecture	

<p>introduces into the computational methods and tools for semantic information processing which have been developed by Artificial Intelligence research. Topics covered include problem solving by heuristic search, constraint solving, search strategies for games, representations for domain-specific knowledge, reasoning with formal ontologies, technologies of the Semantic Web, machine learning and knowledge discovery. The design of intelligent agents and agent systems is adopted as unifying perspective for presenting the material. Applications from different fields such as geographic information systems, digital libraries, and social computing illustrate how the methods from semantic information processing are used to build intelligent assistant systems.</p>	
<p>Literatur: Russell, S., Norvig, P. & Davis, E. (2010): Artificial Intelligence. A Modern Approach. 3rd. Upper Saddle River: Prentice Hall. Hitzler, P.; Krötzsch, M.; Rudolph, S. (2010): Foundations of Semantic Web technologies. CRC Press</p>	
<p>2. Semantic Information Processing Lab Lehrformen: Übung Dozenten: Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften Sprache: Englisch</p> <hr/> <p>Inhalte: The course applies the concepts and methods taught in the lecture by solving practical exercises. Most of the exercises can be completed with paper and pencil while some include programming in Java or working with software tools for semantic information processing. The solutions to the exercises are prepared as homework and presented by the students during the lab sessions.</p> <hr/> <p>Literatur: see lecture</p>	<p>2,00 SWS</p>
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten Beschreibung: The written exam covers the material presented in the lecture and the lab sessions.</p>	

Modul KogSys-IA-B Intelligente Agenten <i>Intelligent Agents</i>		6 ECTS / 180 h
Version 1.0.0 (seit SS13) Modulverantwortliche/r: Ute Schmid		
Inhalte: Die Veranstaltung vermittelt grundlegendes Wissen und Kompetenzen im Bereich "Kognitiv orientierte Künstliche Intelligenz" mit Fokus auf Problemlösen und Planung.		
Lernziele/Kompetenzen: Die Studierenden können: <ul style="list-style-type: none"> • grundlegende Forschungsziele der Künstlichen Intelligenz nennen • grundlegende Forschungsfragen der Künstlichen Intelligenz erläutern • gegebene Planungsprobleme in formalen Sprachen modellieren • zentrale formale Methoden des Problemlösens, des Planens und des deduktiven Schließens nennen und auf gegebene Problemstellungen anwenden • Planungssysteme hinsichtlich ihrer Einsatzmöglichkeiten vergleichen • aktuelle Forschungsergebnisse im Bereich Action Planning analysieren und bewerten 		
Bemerkung: Veranstaltung Deutsch (im Bedarfsfall English). Die Folien sowie weitere Materialien sind überwiegend in englischer Sprache. Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: 22.5 h Vorlesung + 30 h Nachbereitung über 15 Wochen 22.5 h Übung + 75 h Bearbeitung von Übungsaufgaben über 15 Wochen 30 h Klausurvorbereitung		
Empfohlene Vorkenntnisse: Kenntnisse entsprechend den Modulen Gdl-MfI-1 (Mathematik für Informatiker) und MI-AuD-B (Algorithmen und Datenstrukturen) oder des Moduls KogSys-KogInf-Psy.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Intelligente Agenten Lehrformen: Vorlesung Dozenten: Ute Schmid Sprache: Deutsch/Englisch Inhalte: In der Vorlesung werden wesentliche Konzepte und Methoden der kognitiv orientierten Künstlichen Intelligenz mit dem Fokus auf Problemlösen und Planen	2,00 SWS

<p>eingeführt. Wesentliche Themengebiete sind: STRIPS-Planung, Logik und Deduktives Planen, heuristische Suche und heuristisches Planen, Planning Graph Techniken, SAT-Planning, Multiagenten-Planung, Bezüge zum menschlichen Problemlösen und Planen.</p> <p>Literatur: Russell & Norvig: Artificial Intelligence -- A Modern Approach; Ghallab, Nau, Traverso: Automated Planning; Wooldridge: An Introduction to Multiagent Systems; Schöning: Logik für Informatiker; Sterling, Shapiro: Prolog</p>	
<p>2. Übung Intelligente Agenten Lehrformen: Übung Dozenten: Michael Siebers Sprache: Deutsch/Englisch</p> <p>Inhalte: Vertiefung von in der Vorlesung eingeführten Methoden und Techniken, zum Teil mit Programmieraufgaben in PROLOG.</p> <p>Literatur: siehe Vorlesung</p>	<p>2,00 SWS</p>
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: In der Klausur können 90 Punkte erzielt werden. Die Klausur ist bestanden, wenn mindestens 40 Punkte erreicht werden.</p> <p>Im Semester werden Übungsblätter ausgegeben für deren freiwillige Bearbeitung eine bzw. zwei Wochen zur Verfügung stehen. Die Lösung der Übungsblätter wird bewertet. Bei bestandener Klausur wird die Bewertung der Übungsblätter für die Berechnung der Note mit berücksichtigt. Eine 1.0 ist dabei auch ohne Punkte aus den Übungsblättern erreichbar.</p> <p>Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialien, Taschenrechner</p>	

Modul KTR-Datkomm-B Datenkommunikation		6 ECTS / 180 h
<i>Data communication</i>		
Version 1.0.0 (seit WS10/11)		
Modulverantwortliche/r: Prof. Dr. Udo Krieger		
<p>Inhalte:</p> <p>Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Dienstangebots. Es werden die geläufigsten Dienste-, Netz- und Protokollarchitekturen öffentlicher und privater Datenkommunikationsnetze wie das OSI-Referenzmodell bzw. die TCP/IP-Protokollfamilie mit aufgesetzten Dateitransfer, World Wide Web und Multimedia-Diensten vorgestellt.</p> <p>Ferner werden die Grundprinzipien der eingesetzten Übertragungsverfahren, der Übertragungssicherungs- und Steuerungsalgorithmen und der wichtigsten Medienzugriffsverfahren diskutiert, z.B. geläufige Übertragungs- und Multiplexverfahren wie FDMA, TDMA, CDMA, Medienzugriffstechniken der CSMA-Protokollfamilie inklusive ihrer Umsetzung in LANs nach IEEE802.x Standards, Sicherungsprotokolle der ARQ-Familie sowie Flusskontrollstrategien mit variablen Fenstertechniken und ihre Realisierung im HDLC-Protokoll.</p> <p>Außerdem werden grundlegende Adressierungs- und Vermittlungsfunktionen in Rechnernetzen wie Paketvermittlung in Routern und Paketverkehrslenkung nach Kürzeste-Wege-Prinzipien bzw. Verkehrslenkung nach dem Prinzip virtueller Wege dargestellt. Darüber hinaus werden die Grundfunktionen der Transportschicht und ihre exemplarische Umsetzung in TCP erläutert.</p>		
<p>Lernziele/Kompetenzen:</p> <p>Die Studierenden sollen zu eigenständigem Arbeiten im Bereich moderner Kommunikationsnetze befähigt werden. Es werden Grundkenntnisse der Datenkommunikation und die systematische Analyse der verwendeten Algorithmen mit Hilfe eines interaktiven Konzeptes theoretischer und praktischer Übungsaufgaben vermittelt. Die Studierenden lernen, gegebene Implementierungen der vorgestellten Datenkommunikationsverfahren zu analysieren und durch Messungen im Kommunikationslabor ihr Leistungsverhalten zu überprüfen.</p>		
<p>Bemerkung:</p> <p>Der Arbeitsaufwand gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> • Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden • Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden • Prüfungsvorbereitung: 35 Stunden 		
<p>Empfohlene Vorkenntnisse:</p> <ul style="list-style-type: none"> • erfolgreich abgeschlossene Prüfungen der Grundlagenfächer des Bachelorstudiums, insbesondere Einführung in die Informatik sowie Algorithmen und Datenstrukturen • gute Programmierkenntnisse in JAVA (oder C++) <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen</p>		<p>Besondere Bestehensvoraussetzungen:</p> <p>Siehe Prüfungsordnung</p>
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls:

		1 Semester
Lehrveranstaltungen		
1. Vorlesung Datenkommunikation Lehrformen: Vorlesung Dozenten: Prof. Dr. Udo Krieger Sprache: Deutsch/Englisch		2,00 SWS
Inhalte: Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Dienstangebots. Es werden die geläufigsten Dienste-, Netz- und Protokollarchitekturen öffentlicher und privater Datenkommunikationsnetze wie das OSI-Referenzmodell bzw. die TCP/IP-Protokollfamilie mit aufgesetzten Dateitransfer, World Wide Web und Multimedia-Diensten vorgestellt. Ferner werden die Grundprinzipien der eingesetzten Übertragungs-, Übertragungssicherungs- und Steuerungsalgorithmen und des Medienzugriffs diskutiert, z.B. geläufige Übertragungs- und Multiplextechniken wie FDMA, TDMA und CDMA Medienzugriffstechniken der CSMA-Protokollfamilie inklusive ihrer Umsetzung in LANs nach IEEE802.x Standards, Sicherungsprotokolle der ARQ-Familie sowie Flusskontrollstrategien mit variablen Fenstertechniken und ihre Realisierung. Außerdem werden grundlegende Adressierungs- und Vermittlungsfunktionen in Rechnernetzen wie Paketvermittlung in Routern und Paketverkehrlenkung dargestellt. Darüber hinaus werden die Grundfunktionen der Transportschicht und ihre exemplarische Umsetzung in TCP erläutert.		
Literatur: <ul style="list-style-type: none"> • Lean-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2004 • Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 4. Aufl., 2003 • Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014 • Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2004 Weitere Angaben und Erläuterungen erfolgen in der 1. Vorlesung.		
2. Übung Datenkommunikation Lehrformen: Übung Dozenten: Prof. Dr. Udo Krieger, Mitarbeiter Informatik, insbesondere Kommunikationsdienste, Telekommunikationssysteme und Rechnernetze Sprache: Deutsch/Englisch		2,00 SWS
Inhalte: Es werden Grundkenntnisse der Datenkommunikation und die systematische Analyse der dabei verwendeten Algorithmen mit Hilfe eines interaktiven Übungskonzeptes aus Haus- und Laboraufgaben vermittelt. Vorlesungsbegleitend werden diese Übungsaufgaben zu folgenden Themen bearbeitet:		

- Netzentwurfsprinzipien
- OSI-Protokolle
- TCP/IP-Protokollstapel
- Netzelemente
- Datenübertragungssicherungsschicht
- Medienzugriffsschicht

Die Studierenden lernen, gegebene Implementierungen der vorgestellten

Datenkommunikationsverfahren mathematisch und kommunikationstechnisch zu analysieren, durch Messungen ihr Leistungsverhalten zu überprüfen und Vor- bzw. Nachteile der Lösungen zu bewerten.

Im Verlauf des Semesters können durch die Bearbeitung der Laboraufgaben

der Übung und die erfolgreiche Bewertung der entsprechenden Teilleistungen eine maximale Anzahl von Bonuspunkten erworben werden. Diese Bonuspunkte werden bei der Notenvergabe des Moduls berücksichtigt. Die Berechnungs-, Vergabe- und Anrechnungsmodalitäten der Bonuspunkteregelung werden zu Beginn der Lehrveranstaltung festgelegt und den Studierenden zur Kenntnis gebracht. Diese Bonuspunkte stellen eine freiwillige Zusatzleistung dar. Das Bestehen der Modulprüfung ist grundsätzlich ohne diese Zusatzleistung möglich. Das Erreichen der Note 1.0 ist ebenfalls ohne die Erbringung dieser Zusatzleistung möglich.

Literatur:

- Lean-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2004
- Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 4. Aufl., 2003
- Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014
- Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2004

Weitere Literatur wird in der Übung benannt.

Prüfung

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

Beschreibung:

Die Inhalte der Vorlesung sowie die Aufgabenstellungen, Lösungen und Erkenntnisse der Übung, die Haus- und Laboraufgaben beinhaltet, werden in Form einer Klausur geprüft.

Im Verlauf des Semesters können durch die Bearbeitung der Laboraufgaben der Übung und die erfolgreiche Bewertung der entsprechenden Teilleistungen eine maximale Anzahl von Bonuspunkten erworben werden. Diese Bonuspunkte werden bei der Notenvergabe des Moduls berücksichtigt. Die Berechnungs-, Vergabe- und Anrechnungsmodalitäten der Bonuspunkteregelung werden zu Beginn der Lehrveranstaltung festgelegt und den Studierenden zur Kenntnis gebracht. Diese Bonuspunkte stellen eine freiwillige Zusatzleistung dar. Das Bestehen der Modulprüfung ist grundsätzlich ohne diese Zusatzleistung möglich.

Zulässige Hilfsmittel der Prüfung:

- | | |
|---|--|
| <ul style="list-style-type: none">• Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay | |
|---|--|

Modul KTR-GIK-M Grundbausteine der Internet-Kommunikation <i>Foundations of Internet Communication</i>	6 ECTS / 180 h
Version 1.0.0 (seit WS10/11) Modulverantwortliche/r: Prof. Dr. Udo Krieger	
Inhalte: Die Lehrveranstaltung vermittelt eine Einführung in die theoretischen Grundlagen wichtiger kommunikationstechnischer Problemstellungen zu den Themengebieten Grundlagen der Internet-Kommunikation, Verbindungssegmente und Routing in IP-Netzen, Transportprotokolle in IP-Netzen bzw. fortgeschrittener Module wie Echtzeit-Kommunikation und Sicherheit in IP-Netzen und die eigenständige praktische Umsetzung des erworbenen Wissens durch vorgegebene Laborübungen zur Internet-Kommunikation in Kleingruppen. Dabei werden weitere Hilfsmittel und Anleitungen sowie die Laborumgebung bereitgestellt. Zur Implementierung soll ein Rechnernetz im Labor konfiguriert und getestet werden. Die Betriebssystem-Grundausrüstung und erforderliche Software-Werkzeuge wie Wireshark und Atheris werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet.	
Lernziele/Kompetenzen: Wichtige Fertigkeiten zur Bewertung aktueller Kommunikationstechnologien sind nur durch die Vermittlung praktischer Fähigkeiten und Erfahrungen in teamorientierten Prozessen unter Zeit- und Zielvorgaben industrienah erlernbar. Die Studierenden werden in der Vorlesung Grundbausteine der Internet-Kommunikation und den begleitenden Laborübungen zu eigenverantwortlichem, team-orientierten Arbeiten angeleitet. Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten Datenkommunikation und die Fähigkeit, Lösungsvorschläge der modernen Internet-Kommunikation sicher beurteilen zu können. Die Lehrveranstaltung "Grundbausteine der Internet-Kommunikation" hat folgende Zielsetzungen: <ul style="list-style-type: none"> • Fortführung der Vorlesung Datenkommunikation des Bachelorprogrammes als Profilbildungsstudium auf Masterniveau • praktisches Erarbeiten der Grundlagen der Internet- und Multimedia-Kommunikation • Aufbau und Verkehrsanalyse von TCP/IP-basierten Rechnernetzen mit modernen Echtzeit- und Web-Anwendungen • Angebot einer Prüfungsalternative zur Lehrveranstaltung Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen (KTR-MMK-M) oder Mobilkommunikation (KTR-Mobi-M) im Prüfungsfach Kommunikationssysteme und Rechnernetze • Ergänzung der Lehrangebote in Verteilten Systemen und Medieninformatik zur Bildung eines Studienschwerpunktes "Mobile verteilte Systeme" bzw. Next Generation Internet Die Lehrveranstaltung ist für Bachelorstudierende im Profilbildungsstudium zur Stärkung ihrer Arbeitsmarktchancen, für Masterstudierende sowie für Austauschstudenten/innen besonders empfehlenswert.	
Bemerkung: Der Arbeitsaufwand gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Präsenzveranstaltungen (Vorlesung, Laborübungen, Laborbesprechungen): 45 Stunden • Vorbereitung, Ausführung und Nachbereitung von Vorlesungen und Laborübungen: 100 Stunden • Prüfungsvorbereitung: 35 Stunden 	

The module can be selected by exchange students and master students speaking only English.		
Empfohlene Vorkenntnisse: <ul style="list-style-type: none"> • Datenkommunikation im Umfang KTR-Datkomm-B • Programmierkenntnisse in JAVA (oder C++) • der Erwerb von LINUX-Kenntnissen wird empfohlen, ist aber keine Voraussetzung <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen</p>		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Grundbausteine der Internet-Kommunikation Lehrformen: Vorlesung und Übung Dozenten: Prof. Dr. Udo Krieger Sprache: Englisch/Deutsch	4,00 SWS
Inhalte: Die Lehrveranstaltung vermittelt eine Einführung in die theoretischen Grundlagen wichtiger kommunikationstechnischer Problemstellungen zu den Themengebieten Grundlagen der Internet-Kommunikation, Verbindungssegmente und Routing in IP-Netzen, Transportprotokolle in IP-Netzen bzw. fortgeschrittener Module wie Echtzeit-Kommunikation und Sicherheit in IP-Netzen und die eigenständige praktische Umsetzung des erworbenen Wissens durch vorgegebene Laborübungen zur Internet-Kommunikation in Kleingruppen. Dabei werden weitere Hilfsmittel und Anleitungen sowie die Laborumgebung bereitgestellt. Zur Implementierung soll ein Rechnernetz im Labor konfiguriert und getestet werden. Die Betriebssystem-Grundausrüstung und erforderliche Software-Werkzeuge wie Wireshark und Atheris werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet. Die Organisation der Arbeiten erfolgt in einem industrienahen Projektrahmen aus Definitions-, Vorbereitungs-, Implementierungs- und Präsentationsphasen. Dabei soll, wie in realen Projekten üblich, eine inkrementelle Vorgehensweise durchgeführt werden, d.h: <ul style="list-style-type: none"> • Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages), • ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen • und der Darlegung von Zwischenergebnissen bzw. • einem Abschlussbericht mit Abschlusspräsentation Weitere Laboraufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" werden bei Bedarf in die Lehrveranstaltung integriert. Details werden in der Vorlesung angekündigt.	

Eine aktuelle Liste der bearbeiteten Themen der Lehrveranstaltung wird in der Vorlesung bereitgestellt.

Literatur:

Grundlagen:

- J. Liebeherr, M. Elzarki: Mastering Networks, An Internet Lab Manual, Pearson Education, Boston, 2004.

weitere Literatur zu einzelnen Arbeitspaketen:

- Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014 .
- Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 6. Aufl., 2013.
- Sikora, A.: Technische Grundlagen der Rechnerkommunikation, Fachbuchverlag Leipzig, 2003.
- Leon-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2nd ed. 2004.
- Badach, A.: Voice over IP - Die Technik, Carl Hanser Verlag, München, 2. Aufl., 2005.
- Flaig, G., u.a.: Internet-Telefonie, Open source Press, München, 2006.

Eine aktualisierte Liste wird in der Vorlesung bereitgestellt.

Prüfung

Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Monate

Beschreibung:

Die Leistungsbewertung der Lehrveranstaltung erfolgt nach Abschluss auf folgender Grundlage:

- Auswertung des in Gruppenarbeit gemeinsam erstellten schriftlichen Projektberichtes der bearbeiteten Aufgaben (30% der Endbewertung)
- Vorführung und Erläuterungen der Zusammenhänge einzelner Aufgaben und Ergebnisse im Rahmen einer individuellen Kolloquiumsprüfung im Umfang von 30 Minuten (70% der Endbewertung)

Die individuelle Gesamtleistung muss mit der Note "ausreichend" bewertet werden, um die Prüfung zu bestehen.

Modul KTR-Mfi-2 Mathematik für Informatiker 2 (Lineare Algebra)		6 ECTS / 180 h
<i>Mathematics for Computer Science 2 (Linear Algebra)</i>		
Version 2.0.0 (seit WS12/13) Modulverantwortliche/r: Prof. Dr. Udo Krieger		
Inhalte: Die Lehrveranstaltung stellt mathematische Grundlagen der Informatik bereit und ist dem Pflichtbereich der Modulgruppe A1 "Mathematische Grundlagen" für Angewandte Informatik/Software Systems Science zugeordnet. Der besondere Bezug zur Angewandten Informatik bzw. Software Systems Science wird in den Vorlesungsbeispielen und Übungen herausgearbeitet. Es werden grundlegende Methoden und Algorithmen der Gruppen- und Ringtheorie, der linearen Algebra, der Matrizenalgebra, der Lösungstheorie linearer Gleichungssysteme, der Eigenwerttheorie sowie spezifische Anwendungen der Informatik vorgestellt.		
Lernziele/Kompetenzen: Die Studierenden lernen, die grundlegende Methoden und Algorithmen der Lineare Algebra anzuwenden und spezifische Anwendungen der Angewandten Informatik als Probleme der linearen Algebra zu erkennen, zu formulieren und mit Hilfe geeigneter Verfahren zu lösen.		
Bemerkung: Das Modul stellt die Grundlagen für Studierende der Angewandten Informatik und Software Systems Science sowie Studierendende im Nebenfach verwandter Bachelorstudiengänge der Fakultät WIAI bereit. Der Arbeitsaufwand gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden • Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden • Prüfungsvorbereitung: 35 Stunden 		
Empfohlene Vorkenntnisse: Schulkenntnisse auf dem Niveau eines Mathematik-Vorkurs Modul Mathematik-Vorkurs-Bachelorstudium (KTR-MVK-B) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester: 1. - 2.	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Mathematik für Informatiker 2 (Lineare Algebra) Lehrformen: Vorlesung und Übung Dozenten: Prof. Dr. Udo Krieger Sprache: Deutsch	4,00 SWS
Inhalte: Die Lehrveranstaltung stellt mathematische Grundlagen der Informatik bereit und ist dem Pflichtbereich der Modulgruppe A1 "Mathematische Grundlagen" für Angewandte Informatik/Software Systems Science zugeordnet.	

<p>Es werden grundlegende Methoden und Algorithmen der Gruppen- und Ringtheorie, der linearen Algebra, der Matrizenalgebra, der Lösungstheorie linearer Gleichungssysteme, der Eigenwerttheorie sowie spezifische Anwendungen der Informatik vorgestellt.</p>	
<p>Literatur:</p> <ul style="list-style-type: none">• A. Steger: Diskrete Strukturen 1, Springer, Heidelberg, 2002.• G. Golub, C.F. van Loan: Matrix Computations, 3ed., Johns Hopkins, 1996.• D. Hachenberger: Mathematik für Informatiker, Pearson, 2008.• J. Liesen, V. Mehrmann, Lineare Algebra, Vieweg+Teubner, Bachelorkurs Mathematik, 2011.• B. Pareigis: Linear Algebra für Informatiker, Springer, 2000.• M.P.H. Wolff u.a.: Mathematik für Informatik und Bioinformatik, Springer, 2004.• Weitere Literatur wird in der Vorlesung benannt.	
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: Schriftliche Prüfung zu Inhalten der Vorlesung und Übungen im Umfang von 90 Minuten. Zugelassene Hilfsmittel werden in der Lehrveranstaltung bekannt gegeben.</p>	

<p>Modul KTR-SSSProj-B KTR Bachelorprojekt Software Systems Science <i>KTR Bachelor Project Software Systems Science</i></p>	<p>12 ECTS / 360 h</p>
<p>Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Udo Krieger</p>	
<p>Inhalte: Wichtige Fertigkeiten bei der Anwendung neuer Kommunikationstechnologien und zur Entwicklung neuer Kommunikationsdienste sind nur durch die Vermittlung praktischer Fähigkeiten und Erfahrungen in teamorientierten Prozessen unter Zeit- und Zielvorgaben industrienah erlernbar. Die Studierenden werden in der Lehrveranstaltung in einem angeleiteten, aber ansonsten eigenverantwortlich durchgeführten teamorientierten Arbeitsprozess aktuelle Entwicklungsaufgaben aus dem Forschungsbereich der Professur für Informatik bearbeiten.</p>	
<p>Lernziele/Kompetenzen: Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation. Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten, qualitätsgesicherten Multimediakommunikation und die Fähigkeit, Lösungsvorschläge moderner Dienstarchitekturen im Internet der Zukunft sicher beurteilen zu können. Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Software-Projekten im Bereich Kommunikationsnetze und -dienste auftretenden konzeptionellen und praktischen Probleme wie auch von erfolgsversprechenden Lösungsansätzen dieser Probleme erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der Professur für Informatik in Kleingruppen oder einzeln geschieht, gewinnen die Studierenden wichtige Erfahrungen in der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in wissenschaftlich ausgerichteten Arbeitsberichten und in der professionellen Präsentation dieser Ergebnisse.</p>	
<p>Bemerkung: Dieses Modul erstreckt sich über 2 Semester und umfasst 2x6=12 ECTS und 2x4=8 SWS. Der Arbeitsaufwand beträgt insgesamt 360 Std., welche sich grob wie folgt gliedern: • 30 Std. Recherche, Planung und Teilnahme am Planungsworkshop • 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien • 180 Std. Durchführung des Projekts (Projektarbeit) • 50 Std. Erstellung des Zwischenberichts (Hausarbeit) • 60 Std. Erstellung des Abschlussberichts, Erstellung und Präsentation der Projektergebnisse (Hausarbeit und Kolloquium)</p>	
<p>Empfohlene Vorkenntnisse:</p> <ul style="list-style-type: none"> • gute Kenntnisse in Mathematik für Informatiker 2 • mindestens gute JAVA (oder C/C++) Kenntnisse 	<p>Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung</p>

<ul style="list-style-type: none"> • Kenntnisse der Datenkommunikation im Umfang von KTR-Datkomm-B oder vergleichbare Kenntnisse werden empfohlen • grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z.B. im Umfang des Moduls "Software EngineeringLab" (SWT-SWL-B), werden empfohlen. <p>Modul Programmierung komplexer interagierender Systeme (DSG-PKS-B) - empfohlen</p> <p>Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen</p> <p>Modul Mathematik für Informatiker 2 (Lineare Algebra) (KTR-Mfi-2) - empfohlen</p>		
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester: 4.	Minimale Dauer des Moduls: 2 Semester

Lehrveranstaltungen	
<p>KTR Bachelorprojekt Software Systems Science</p> <p>Lehrformen: Übung</p> <p>Dozenten: Prof. Dr. Udo Krieger</p> <p>Sprache: Englisch/Deutsch</p> <hr/> <p>Inhalte:</p> <p>Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation. Im Mittelpunkt steht die eigenständige, teamorientierte praktische Umsetzung eines Entwicklungsauftrages unter Verwendung des erworbenen Wissens einzelner Lehrveranstaltungen des Fachgebiets der Professur für Informatik.</p> <p>Die Betriebssystem-Grundausstattung und erforderliche Software-Werkzeuge wie Vyatta-Router, Wireshark, Atheris und RapidStream werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet.</p> <p>Die Lehrveranstaltung erstreckt sich über 2 konsekutive Semester. Die Organisation der Arbeiten erfolgt in einem industrienahen Projektrahmen aus Definitions-, Vorbereitungs-, Implementierungs- und Präsentationsphasen. Dabei soll, wie in realen Projekten üblich, eine inkrementelle Vorgehensweise durchgeführt werden, d.h:</p> <ul style="list-style-type: none"> • Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages), • ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen • und der Darlegung von Zwischenergebnissen in einem Zwischenbericht nach dem 1. Semester sowie • einem Abschlussbericht mit Abschlusspräsentation der Arbeitsergebnisse in einem Kolloquium im 2. Semester. <p>Es werden Entwicklungsaufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" bearbeitet. Details werden auf der Webseite der Lehrveranstaltung angekündigt. Eine aktuelle Liste der bearbeiteten Themen der Lehrveranstaltung wird in der 1. Besprechung bereitgestellt.</p>	8,00 SWS

<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 6 Monate Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung Beschreibung: Die Lehrveranstaltung erstreckt sich über 2 konsekutive Semester. Es werden die Leistungen der als Gruppen- oder Einzelarbeit ausgeführten schriftlichen Ausarbeitung der Projektaufgaben und ihrer Präsentation im 2. Semester sowie die Ergebnisse einer abschließenden, individuellen Kolloquiumsprüfung mit einem Anteil von 60% der Gesamtnote bewertet. Hausarbeit und Kolloquium gehen dabei zu gleichen Teilen in die Bewertung ein. Alle Teilleistungen müssen in jedem Semester erfolgreich absolviert werden, um eine finale Anrechnung der Lehrveranstaltung zu erhalten.</p>	
---	--

<p>Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 6 Monate Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung Beschreibung: Die Lehrveranstaltung erstreckt sich über 2 konsekutive Semester. Es werden die Leistungen der als Gruppen- oder Einzelarbeit ausgeführten schriftlichen Ausarbeitung der Projektaufgaben mit einer Bearbeitungsdauer von 6 Monaten im 1. Semester mit einem Anteil von 40% der Gesamtnote bewertet.</p>	
---	--

Modul Mathe-B-01 (BWL) Mathematik für Wirtschaftswissenschaftler I (BWL)	3 ECTS / 90 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Dr. rer. pol. Reinhard Dobbener	
<p>Inhalte:</p> <ul style="list-style-type: none"> 0 Grundlagen 0.1 Kartesische Produkte und Relationen 0.2 Abbildungen 1 Folgen und Reihen <ul style="list-style-type: none"> 1.1 Folgen 1.2 Reihen 1.3 Finanzmathematik <ul style="list-style-type: none"> 1.3.1 Einfache Zinsrechnung 1.3.2 Zinseszinsrechnung 1.3.3 Rentenrechnung 1.3.4 Tilgungsrechnung 1.3.5 Investitionsrechnung 2 Differenzialrechnung <ul style="list-style-type: none"> 2.1 Funktionen einer und mehrerer Variablen <ul style="list-style-type: none"> 2.1.1 Beispiele, grafische Darstellung und Eigenschaften von Funktionen einer und mehrerer Variablen 2.1.2 Polynome, gebrochen rationale und algebraische Funktionen 2.1.3 Transzendente Funktionen (Exponential-, Logarithmus- und Winkelfunktionen) 2.1.4 Grenzwerte und Stetigkeit von Funktionen 2.2 Differenzialrechnung für Funktionen einer Variablen <ul style="list-style-type: none"> 2.2.1 Differenzialquotient und Ableitungsregeln 2.2.2 Differenziation der Grundfunktionen 2.2.3 Monotonie, Konvexität/Konkavität und Extremstellen differenzierbarer Funktionen einer Variablen 2.2.4 Rechnen mit dem Symbol #, die Regeln von de l'Hospital 2.2.5 Approximation differenzierbarer Funktionen durch Polynome, Differenziale und der Satz von Taylor 2.2.6 Elastizitäten 2.3 Differenzialrechnung für Funktionen mehrerer Variablen <ul style="list-style-type: none"> 2.3.1 Partielle und totale Ableitungen 2.3.2 Die Kettenregel für Funktionen mehrerer Variablen 2.3.3 Partielle Ableitungen höherer Ordnung 2.3.4 Partielle und totale Differenziale, partielle Elastizitäten 2.3.5 Implizite Funktionen 2.3.6 Extremstellen differenzierbarer Funktionen mehrerer Variablen (ohne Nebenbedingungen) 2.3.7 Extremstellen differenzierbarer Funktionen mehrerer Variablen (mit Nebenbedingungen) 2.3.8 Differenziation vektorwertiger Funktionen 3 Integralrechnung <ul style="list-style-type: none"> 3.1 Das unbestimmte Integrale 3.2 Das bestimmte Integrale 3.3 Uneigentliche Integrale 3.4 Integralrechnung für Funktionen mehrerer Variablen 	
Lernziele/Kompetenzen:	

<p>Vermittlung von mathematischen Grundkenntnissen aus dem Gebiet der Analysis. Die Teilnehmer dieser Vorlesung/Übung sollen in die Lage versetzt werden, die mathematischen Verfahren und Konzepte der weiterführenden (wirtschafts-)informatischen Veranstaltungen zu verstehen und zu beherrschen.</p>		
<p>Empfohlene Vorkenntnisse: keine</p>		<p>Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung</p>
<p>Angebotshäufigkeit: WS, SS</p>	<p>Empfohlenes Fachsemester:</p>	<p>Minimale Dauer des Moduls: 1 Semester</p>

<p>Lehrveranstaltungen</p>	
<p>Mathematik für Wirtschaftswissenschaftler I Lehrformen: Vorlesung und Übung Dozenten: Dr. rer. pol. Reinhard Dobbener Sprache: Deutsch</p>	<p>3,00 SWS</p>
<p>Literatur:</p> <ul style="list-style-type: none"> • Chiang A.C.: Fundamental Methods of Mathematical Economics, McGraw-Hill, New York, 1967. • Dobbener R.: Analysis - Studienbuch für Ökonomen, 2. Auflage, Oldenbourg, München, Wien, 1993. • Gal T., Kruse H.J., Vogeler B., Wolf H.: Mathematik für Wirtschaftswissenschaftler, Band 1-3, Springer, Berlin, Heidelberg, New York, Tokio, 1983. • Opitz O.: Mathematik, Oldenbourg, München, Wien, 1989. • Schwarze J.: Mathematik für Wirtschaftswissenschaftler, Band 1-3, Neue Wirtschaftsbriefe, Herne, Berlin, 1981. • Sydsaeter K., Hammond P.: Mathematik für Wirtschaftswissenschaftler, Pearson Studium, München 2004. 	

<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten Zulassungsvoraussetzung zur Modulprüfung: keine</p>	
--	--

Modul MI-AuD-B Algorithmen und Datenstrukturen <i>Algorithms and Data Structures</i>		6 ECTS / 180 h
Version 1.0.0 (seit SS07) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
Inhalte: Grundlegende Algorithmen (z. B. Suchen, Sortieren, einfache Graphalgorithmen) und Datenstrukturen (z. B. Listen, Hashtabellen, Bäume, Graphen) werden vorgestellt. Konzepte der Korrektheit, Komplexität und Algorithmenkonstruktion werden betrachtet.		
Lernziele/Kompetenzen: Das Modul vermittelt die Kompetenz, die Qualität von Datenstrukturen und Algorithmen im Hinblick auf konkrete Anforderungen einzuschätzen und ihre Implementierung in einem Programm umzusetzen. Daneben sollen grundlegende Kompetenzen im Bereich der Algorithmenkonstruktion erworben werden. Durch die Übung soll auch Sicherheit im Umgang mit objektorientierten Entwicklungsmethoden und Standardbibliotheken erworben und Teamarbeit geübt werden.		
Bemerkung: Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung) • Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden • Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb) • Bearbeiten der 6 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb) • Klausurvorbereitung und Klausur: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) 		
Empfohlene Vorkenntnisse: Grundlegende Kenntnisse in Informatik und Programmierung, wie sie z. B. im Modul DSG-EiAPS-B vermittelt werden.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester: 2.	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Algorithmen und Datenstrukturen Lehrformen: Vorlesung Dozenten: Prof. Dr. Andreas Henrich Sprache: Deutsch	2,00 SWS
Inhalte: Die Vorlesung betrachtet die klassischen Bereiche des Themengebiets Algorithmen und Datenstrukturen: <ul style="list-style-type: none"> • Einleitung • Listen • Hashverfahren • Bäume 	

<ul style="list-style-type: none"> • Graphen • Sortieren • Algorithmenkonstruktion 	
<p>Literatur: Als begleitende Lektüre wird ein Standardlehrbuch über Algorithmen und Datenstrukturen empfohlen. Beispiele wären:</p> <ul style="list-style-type: none"> • Saake, Gunter; Sattler, Kai-Uwe: Algorithmen und Datenstrukturen: Eine Einführung mit Java, ISBN: 978-3-89864-385-6, 3. Aufl. 2006, 512 Seiten, Dpunkt Verlag • Ottmann, Thomas; Widmayer, Peter: Algorithmen und Datenstrukturen, ISBN: 978-3-8274-1029-0, 4. Aufl. 2002, 736 Seiten, Spektrum, Akedemischer Verlag 	
<p>2. Übung Algorithmen und Datenstrukturen Lehrformen: Übung Dozenten: Mitarbeiter Medieninformatik Sprache: Deutsch</p> <hr/> <p>Inhalte: In der Übung werden folgende Aspekte betrachtet:</p> <ul style="list-style-type: none"> • Verständnis und Nutzung von Algorithmen • Aufwandsbestimmung für Algorithmen • Implementierung von Algorithmen und Datenstrukturen • Nutzung von Bibliotheken • Anwendung von Prinzipien zur Algorithmenkonstruktion <hr/> <p>Literatur: siehe Vorlesung</p>	<p>2,00 SWS</p>

<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: Gegenstand der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten).</p> <p>In der Klausur können 90 Punkte erzielt werden.</p> <p>Zusätzlich zur Prüfungsdauer wird eine Lesezeit von 15 Minuten gewährt, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.</p> <p>Im Semester werden studienbegleitend 6 Teilleistungen zur freiwilligen Bearbeitung ausgegeben. Für jede Teilleistung stehen in der Regel 2 Wochen als Bearbeitungszeit zur Verfügung. Die Lösungen zu den Teilleistungen werden bewertet. Pro Teilleistung können maximal 2 Punkte erzielt werden. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte (maximal 12 Punkte) als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar.</p>	
---	--

Modul MI-LA-DatSchu-B Grundlagen und Fallstudien zum Datenschutz <i>Foundations and Case Studies on Data Protection</i>		4 ECTS / 120 h
Version 2.0.0 (seit WS10/11) Modulverantwortliche/r: Dr. theol. M.A. phil. Wolfgang Hübner		
Inhalte: Die Anforderungen zum Datenschutz sind in entsprechenden Bundes- und Landesgesetzen niedergelegt. Das Modul bietet Studierenden die Möglichkeit, sich diesem Thema in einem konstruktiven Ansatz zu stellen und die entsprechenden Anforderungen sowie die Möglichkeiten zu ihrer Erfüllung kennenzulernen.		
Lernziele/Kompetenzen: Vermittlung der erforderlichen fachlichen Kenntnisse, Methoden und Fähigkeiten, um die inhaltlichen, organisatorischen und technischen Anforderungen des Datenschutzes und der Datensicherheit in einem Unternehmen umsetzen zu können. Kenntnis der Grundprinzipien des Datenschutzes und der Datensicherheit, der gesetzlichen Anforderungen und der datenschutzrelevanten Rechtsprechung.		
Bemerkung: Der typische Aufwand zum Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • 45 Stunden für den Besuch der Veranstaltung • 60 Stunden für die Nachbereitung und die Betrachtung von Fallstudien • 15 Stunden Prüfungsvorbereitung und Prüfung 		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Grundlagen und Fallstudien zum Datenschutz Lehrformen: Vorlesung und Übung Dozenten: Dr. theol. M.A. phil. Wolfgang Hübner Sprache: Deutsch	4,00 SWS
Inhalte: Gliederung der Veranstaltung <ol style="list-style-type: none"> 1. Ziel des Datenschutzes 2. Grundlagen des BDSG 3. Allgemeine Vorschriften des BDSG 4. Datenschutz im nicht-öffentlichen Bereich 	
Literatur: wird in der Veranstaltung bekannt gegeben	

Prüfung	
----------------	--

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten	
--	--

Modul MI-WebT-B Web-Technologien <i>Web Technologies</i>		6 ECTS / 180 h
Version 2.0.0 (seit SS12) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
Inhalte: Nach eine Betrachtung der Grundlagen werden die verschiedenen Ebenen der Entwicklung von Web-Anwendungen von HTML und CSS über JavaScript und entsprechende Bibliotheken bis hin zur Serverseite und Frameworks oder Content Management Systemen betrachtet. Aspekte der Sicherheit von Web-Anwendungen werden ebenfalls angesprochen.		
Lernziele/Kompetenzen: Studierende sollen methodische, konzeptuelle und praktische Fähigkeiten und Fertigkeiten zur Erstellung von Web-Applikationen erwerben. Besonderes Augenmerk wird dabei auf Web 2.0 Technologien gelegt. Die Studierenden sind nach Abschluss des Moduls in der Lage, Web-Anwendungen selbständig mit gängigen Frameworks und Techniken zu entwickeln.		
Bemerkung: Die Lehrveranstaltungen werden in Deutsch durchgeführt. Zahlreiche Quellen und Dokumentationen der Systeme sind aber auf Englisch . Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> • Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung) • Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden • Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb) • Bearbeiten der 3 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb) • Prüfungsvorbereitung und Prüfung: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) 		
Empfohlene Vorkenntnisse: Grundkenntnisse der Informatik und zu Dateiformaten, wie Sie z. B. in den unten angegebenen Modulen erworben werden können. Insbesondere sind auch Kenntnisse in einer imperativen oder objektorientierten Programmiersprache erforderlich. Modul Einführung in die Informatik (DSG-Eidl-B) - empfohlen Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Web-Technologien Lehrformen: Vorlesung Dozenten: Prof. Dr. Andreas Henrich Sprache: Deutsch Inhalte:	2,00 SWS

<p>Die Veranstaltung betrachtet die Aufgabenfelder, Konzepte und Technologien zur Entwicklung von Web-Anwendungen. Folgende Bereiche bilden dabei die Schwerpunkte der Veranstaltung:</p> <ul style="list-style-type: none"> • Das Web: Einführung, Architektur, Protokoll ... • Sprachen zur Beschreibung von Webseiten: HTML & CSS • Client-Side Scripting: die Basics & AJAX • Server-Side Scripting: CGI + PHP • Frameworks • Sicherheit von Web-Anwendungen • CMS, LMS, SEO & Co. 	
<p>Literatur: aktuelle Literatur wird in der Veranstaltung bekannt gegeben</p>	
<p>2. Übung Web-Technologien Lehrformen: Übung Dozenten: Mitarbeiter Medieninformatik Sprache: Deutsch</p> <hr/> <p>Inhalte: praktische Aufgaben zum Stoff der Vorlesung</p> <hr/> <p>Literatur: siehe Vorlesung</p>	<p>2,00 SWS</p>
<p>Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung: Gegenstand der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten). In der Klausur können 90 Punkte erzielt werden. Zusätzlich zur Prüfungsdauer wird eine Lesezeit von 15 Minuten gewährt, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können. Im Semester werden studienbegleitend 3 Teilleistungen zur freiwilligen Bearbeitung ausgegeben. Für jede Teilleistung stehen in der Regel 4 Wochen als Bearbeitungszeit zur Verfügung. Die Lösungen zu den Teilleistungen werden bewertet. Pro Teilleistung können maximal 4 Punkte erzielt werden. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte (maximal 12 Punkte) als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar.</p>	

Modul MOBI-CONF Scientific Conference Management <i>Scientific Conference Management</i>		3 ECTS / 90 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
Inhalte: Scientific conferences are complex real-world events. In this module, students will learn how to organize such an event. The conference to organize will be held at the University of Bamberg and will contain several other seminars.		
Lernziele/Kompetenzen: The students will know how to organize a complex real-world event and how a scientific conference works		
Bemerkung: The project report (schriftliche Hausarbeit) of 8 pages will be related to a successfully organized conference.		
Empfohlene Vorkenntnisse: Wissenschaftliches Arbeiten / Scientific Methods (z.B. durch Modul IAI-WAI-B)		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Seminar Scientific Conference Management Lehrformen: Seminar Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch	2,00 SWS
Inhalte: In the conference organization project, the following main tasks need to be done: <ul style="list-style-type: none"> · create and maintain a web site for the conference · raising and planning budget · prepare one or multiple call for papers · gather a programme committee · organize the review process · organize and book a venue, including the social event · edit the proceedings 	
Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 4 Monate	

Modul MOBI-DSC Data Streams and Complex Event Processing <i>Data Streams and Complex Event Processing</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
Inhalte: The management of data streams and foundations of event processing: applications, systems, query languages, continuous query processing, and security in distributed data stream management systems.		
Lernziele/Kompetenzen: The students will understand the management and processing of data from of active data sources like sensors, social media (e.g., Twitter) or financial transactions.		
Empfohlene Vorkenntnisse: Grundlegende Kenntnisse über relationale Datenbanken, relationale Algebra und SQL, z.B. erworben im Modul SEDA-DMS-B: Datenmanagementsysteme		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Data Streams and Complex Event Processing Lehrformen: Vorlesung Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch <hr/> Inhalte: The lecture covers the following topics: <ul style="list-style-type: none"> · Architectures of data stream management systems · Query languages · Data stream processing · Complex event processing · Security in data stream management systems · Application of data stream management systems 	2,00 SWS
2. Übung Data Streams and Complex Event Processing Lehrformen: Übung Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch <hr/> Inhalte: Siehe Vorlesung	2,00 SWS
Prüfung mündliche Prüfung / Prüfungsdauer: 30 Minuten	

Modul MOBI-IMP-B Implementation of Data Management Systems <i>Implementation of Data Management Systems</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
Inhalte: This modul covers the realization of database systems.		
Lernziele/Kompetenzen: The students will know how a database system can retrieve large amounts of structured data in low latency, and they will be able to understand and apply various indexing strategies for other data management related tasks.		
Empfohlene Vorkenntnisse: Grundlegende Kenntnisse über relationale Datenbanken, relationale Algebra und SQL, z.B. erworben im Modul SEDA-DMS-B: Datenmanagementsysteme		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
Lehrveranstaltungen		
1. Vorlesung Implementation of Data Management Systems Lehrformen: Vorlesung Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch		2,00 SWS
Inhalte: The lecture covers the following topics: <ul style="list-style-type: none"> · Architecture of Database Systems · Storage Structures, Files and Segments · DB Buffer Management · Various indexes and access path methods (hash-based, hierarchical, multi-dimensional) · DB Interfaces (record-oriented, set-oriented) · Implementation of table operations 		
2. Übung Implementation of Data Management Systems Lehrformen: Übung Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch		2,00 SWS
Inhalte: Siehe Vorlesung		
Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten		

Modul MOBI-MSS-B Mobility in Software Systems <i>Mobility in Software Systems</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS15/16) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
Inhalte: This moduls covers architectures, implementation techniques and algorithms for mobile software systems and software systems that manage mobility. This includes client-side aspects (mobile applications like location-based services), server-side aspects (data management of moving objects), and aspects of distribution (data communication). In addition, since many mobile software systems deal with sensitive information like the location of users, aspects of location privacy are covered.		
Lernziele/Kompetenzen: The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such sysetems.		
Empfohlene Vorkenntnisse: Grundlegende Kenntnisse über relationale Datenbanken, relationale Algebra und SQL, z.B. erworben im Modul SEDA-DMS-B: Datenmanagementsysteme		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Mobility in Software Systems Lehrformen: Vorlesung und Übung Sprache: Deutsch/Englisch	4,00 SWS 6 ECTS
Lernziele: The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such sysetems.	
Inhalte: This moduls covers architectures, implementation techniques and algorithms for mobile software systems and software systems that manage mobility. This includes client-side aspects (mobile applications like location-based services), server-side aspects (data management of moving objects), and aspects of distribution (data communication). In addition, since many mobile software systems deal with sensitive information like the location of users, aspects of location privacy are covered.	

Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten Gewicht: 0/NaN	
---	--

Modul MOBI-PRS-B Bachelor Project Mobile Software Systems (SoSySc) <i>Bachelor Project Mobile Software Systems (SoSySc)</i>		12 ECTS / 360 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
Inhalte: Ein überschaubares Projekt zu Themen aus der aktuellen Forschungsarbeit auf den Gebieten Mobile Software Systeme und Datenstromverarbeitung wird in zum Teil gemeinsam, zum Teil arbeitsteilig arbeitenden Kleingruppen von Studierenden - und ggf. auch einzeln - von der Konzeption bis zur Umsetzung und deren Dokumentation und Präsentation durchgeführt. Dabei geht es insbesondere um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe. Dies erfordert eine umfangreiche Recherche aktueller Literatur und zum Thema vorgeschlagener Ansätze, sowie eine detaillierte Planung, deren Ergebnisse in einem Planungsworkshop vorgestellt und diskutiert werden. Ein Beispiel für ein mögliches Projektthema ist Konzeption, Entwicklung und Evaluation eines ortsbasierten, mobilen Spiels oder eines mobilen Stadtinformationssystems.		
Lernziele/Kompetenzen: Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Softwaresystem-Projekten auftretenden konzeptionellen und praktischen Probleme wie auch von erfolgsversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich von mobilen Softwaresystemen in Kleingruppen – oder ggf. auch einzeln – geschieht, gewinnen die Studierenden wichtige Erfahrungen in der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in wissenschaftlich ausgerichteten Arbeitsberichten und in der Präsentation dieser Ergebnisse.		
Bemerkung: Dauer der Lehranstaltung 2 Semester		
Empfohlene Vorkenntnisse: Programmierkenntnisse sowie grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z. B. erworben im Modul "Software Engineering Lab" (SWT-SWL-B), und zum wissenschaftlichen Arbeiten, z. B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik" (IAIWAI-B).		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 2 Semester

Lehrveranstaltungen	
Übung Bachelor project Mobile Software Systems (SoSySc) Lehrformen: Übung Dozenten: Prof. Dr. Daniela Nicklas Sprache: Deutsch/Englisch	8,00 SWS
Inhalte: Durchführung des Projekts.	

<p>Begleitend werden Tutorien angeboten, innerhalb derer die Studierenden mit themenrelevanten Methoden, Techniken und Softwarewerkzeugen vertraut gemacht werden. Darüber hinaus finden regelmäßige Treffen zwischen TeilnehmerInnen/Teilnehmern und Projektbetreuern statt, indem der Projektfortschritt kontrolliert und fachlich diskutiert wird.</p> <p>Der Fortschritt bzw. die Ergebnisse des Projekts werden von Studierenden in einem Zwischen- und Abschlussbericht dokumentiert. Zudem wird von jeder Kleingruppe – bzw. bei Individualprojekten von der Teilnehmerin bzw. dem Teilnehmer – ein Poster zum Projekt und dessen Ergebnissen erstellt, welches im Rahmen einer Abschlussveranstaltung präsentiert, erläutert und diskutiert wird.</p>	
<p>Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 6 Monate</p> <p>Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung</p> <p>Beschreibung: Production of a written report on the software project carried out (Assignment/Hausarbeit).</p>	

<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 6 Monate</p> <p>Beschreibung: Production of a written report on the software project carried out (Assignment/Hausarbeit). Discussion of this project report and of the developed artefacts in the context of the wider project topic (Colloquium/Kolloquium).</p>	
--	--

Modul SEDA-DMS-B Datenmanagementsysteme		6 ECTS / 180 h
Version 1.0.0 (seit WS09/10) Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
Inhalte: Das Modul vermittelt eine systematische Einführung in das Gebiet der Datenmanagementsysteme.		
Lernziele/Kompetenzen: Die Studierenden verstehen die Datenverwaltung auf der Basis des Relationenmodells und kennen grundlegende Architekturkonzepte für Datenmanagementsysteme. Sie erlernen methodische Grundlagen der konzeptuellen Datenmodellierung und verstehen dadurch in vertiefter Weise die Modellierung mit ERM und SERM. Die Studierenden erlernen die Grundlagen der Sprache SQL und können mit SQL Datenbankschemata generieren sowie zugehörige Datenbanken aufbauen und manipulieren. Schließlich sammeln sie erste Erfahrungen im Umgang mit realen Datenbankverwaltungssystemen.		
Empfohlene Vorkenntnisse: Kenntnisse aus dem Modul Grundlagen betrieblicher Informationssysteme sind wünschenswert, jedoch nicht Voraussetzung.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Datenmanagementsysteme Lehrformen: Vorlesung Dozenten: Prof. Dr. Elmar J. Sinz Sprache: Deutsch	2,00 SWS
Inhalte: Datenmanagementsysteme sind zentrale Teilsysteme betrieblicher Anwendungssysteme. Ihre Entwicklung und ihr Betrieb stellen Kernaufgaben der Wirtschaftsinformatik dar. Das Modul vermittelt eine systematische Einführung in diesen Themenbereich. Der Fokus liegt dabei auf der Analyse, der Gestaltung und der Nutzung von Datenmanagementsystemen, nicht etwa auf der Implementierung von Datenbankverwaltungssystemen. Inhaltliche Schwerpunkte bilden das Relationenmodell, die Sprache SQL, Architekturen von Datenmanagementsystemen, der Entwurf von Datenbankschemata, theoretische Grundlagen der Datenmodellierung, Transaktionen und Transaktionsverwaltung sowie der Betrieb von Datenmanagementsystemen. Praktische Fertigkeiten werden insbesondere in Bezug auf den Entwurf von Datenbankschemata und SQL vermittelt. SQL wird anhand von konkreten Datenbankverwaltungssystemen geübt. Fertigkeiten werden insbesondere in Bezug auf SQL vermittelt. Inhalte: <ul style="list-style-type: none"> • Einführung 	

<ul style="list-style-type: none"> • Das Relationenmodell • Die Sprache SQL • Architekturen von Datenmanagementsystemen • Entwurf von Datenbankschemata • Fallstudie: Entwicklung eines Datenmanagementsystems • Theoretische Grundlagen der Datenmodellierung • Transaktionen und Transaktionsverwaltung • Betrieb von datenbankbasierten AWS • Alternative Entwicklungen im Bereich Datenbanken 	
<p>Literatur:</p> <ul style="list-style-type: none"> • Date C.J.: An Introduction to database systems. 8th Edition, Addison-Wesley, Reading, Massachusetts 2003 • Ferstl O.K., Sinz E.J.: Grundlagen der Wirtschaftsinformatik. 7. Auflage, Oldenbourg, München 2012, Kapitel 9.2 • Kemper A., Eickler A.: Datenbanksysteme. Eine Einführung. 8. Auflage, Oldenbourg, München 2011 • Pernul G., Unland R.: Datenbanken im Unternehmen. Analyse, Modellbildung und Einsatz. 2. Auflage, Oldenbourg, München 2003 • Coronel C., Morris S., Rob P.: Database Systems. Design, Implementation, and Management. 9th Edition, Course Technology, Thomson Learning, Boston 2009 • Vossen G.: Datenbankmodelle, Datenbanksprachen und Datenbankmanagement-Systeme. 5. Auflage, Oldenbourg, München 2008 	
<p>2. Übung Datenmanagementsysteme</p> <p>Lehrformen: Übung</p> <p>Dozenten: Mitarbeiter Wirtschaftsinformatik, insb. Systementwicklung und Datenbankanwendung</p> <p>Sprache: Deutsch</p> <hr/> <p>Inhalte:</p> <p>Die Inhalte der Vorlesung werden anhand von Übungsaufgaben und Fallbeispielen vertieft. Praktische Übungen werden unter Verwendung eines gängigen Datenbankverwaltungssystems durchgeführt.</p> <hr/> <p>Literatur:</p> <p>siehe Vorlesung</p>	<p>2,00 SWS</p>
<p>Prüfung</p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

Modul SEDA-EuU-B Entrepreneurship und Unternehmensgründung		3 ECTS / 90 h
Version 1.0.0 Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
Inhalte: Die Veranstaltung vermittelt einen Überblick zum Themengebiet Unternehmensgründung. Schwerpunkt der Veranstaltung ist die Erstellung eines eigenen Businessplans im Verlauf des Semesters. Der Businessplan und insbesondere die eingeschlossene Finanzplanung dienen als Entscheidungsgrundlage pro oder contra Gründung des Unternehmens, indem sie die geplante wirtschaftliche Entwicklung und somit die Tragfähigkeit des Vorhabens aufzeigen.		
Lernziele/Kompetenzen: <ul style="list-style-type: none"> • Studierende können Grundzüge eines Business Plans beschreiben und darstellen. • Studierende können einige nationale Fördermöglichkeiten für unternehmerische Selbständigkeit charakterisieren. • Studierende können ihre Geschäftsidee in einem Business Plan zusammenfassen. • Studierende übernehmen Verantwortung für Prozesse und Produkte des Arbeitens und Lernens in Kleingruppen. • Studierende reflektieren ihre Vorgehensweise bei Lehren und Lernen alleine und in einem gruppenbezogenen Kontext. • Studierende reflektieren ihre Fähigkeiten zur unternehmerischen Selbständigkeit. 		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Entrepreneurship und Unternehmensgründung Lehrformen: Vorlesung und Übung Dozenten: Dr. Markus Wolf Sprache: Deutsch	2,00 SWS
Inhalte: Es werden folgende Punkte eines Businessplans diskutiert: <ul style="list-style-type: none"> • Executive Summary • Kundennutzen und Alleinstellungsmerkmal • Markt & Wettbewerbsanalysen • Marketing & Vertrieb • Geschäftsmodell • Chancen & Risiken • Realisierungsfahrplan • Das Unternehmerteam • Finanzplanung und Finanzierung des Unternehmens 	

Prüfung

Hausarbeit mit Referat / Prüfungsdauer: 20 Minuten

Bearbeitungsfrist: 14 Wochen

Beschreibung:

Die Hausarbeit beinhaltet die Erstellung eines Businessplans.

Die Gewichtung der Prüfungsleistungen Hausarbeit und Referat wird zu Beginn der Lehrveranstaltung von der Dozentin bzw. dem Dozenten bekannt gegeben.

Modul SEDA-PT-B Methoden der Präsentation, Gesprächsführung und Diskussion		3 ECTS / 90 h
Version 1.0.0 Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
Inhalte: An Beispielen von Präsentationen, Einzelgesprächen und Diskussionen sollen <ul style="list-style-type: none"> • persönliche Wirkung auf einzelne und Gruppen • formale und gruppensdynamische Abläufe und • inhaltliche Darstellungsformen bewusst gemacht und zielbezogen für Präsentationen, für Gespräche und für Diskussionen geübt werden.		
Lernziele/Kompetenzen: Die persönliche Wirkung auf Einzelpersonen und Gruppen kennen lernen und verbessern; Inhalte sachlich verständlich, didaktisch ansprechend und adressatengerecht präsentieren; Kurzvorträge, Gespräche und Diskussionen führen und trainieren.		
Bemerkung: Das Modul wird als Blockveranstaltung abgehalten.		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Methoden der Präsentation, Gesprächsführung und Diskussion Lehrformen: Vorlesung und Übung Dozenten: Dr. Ulrich Jentzsch Sprache: Deutsch	2,00 SWS
Lernziele: Lernziele de	
Inhalte: Das Seminar ist als Training konzipiert. Methodisch kommen Einzel- und Gruppenübungen sowie Gruppenarbeiten zur Anwendung. Die persönlichen Verhaltensaspekte werden durch Videoaufzeichnungen dokumentiert und anschließend kommentiert.	

Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten	
--	--

Modul SEDA-TA-B Technikfolgeabschätzung / -bewertung		3 ECTS / 90 h
Version 1.0.0 Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
Inhalte: Das besondere Augenmerk liegt auf der untrennbaren Verflechtung von Naturwissenschaft, Technik und Wirtschaft mit ihren Entwicklungsimpulsen einerseits und der Bedeutung der zum Teil konträren weltanschaulichen Überzeugungen von Bevölkerungsgruppen andererseits. Dieses Spannungsverhältnis unterliegt der Technikfolgenbewertung vor allem: <ul style="list-style-type: none"> • durch das internationale Engagement der Unternehmen, • den immer weniger widerspruchlos akzeptierten Folgen der technisch-wirtschaftlichen Entwicklungen, • sowie dem Handikap, komplexe Prozesse mit weltanschaulichen Aspekten nicht durch streng wissenschaftliche Methoden erfassen zu können. 		
Lernziele/Kompetenzen: Naturwissenschaften, Technik und Wirtschaft haben derzeit wohl den größten Einfluss auf das Denken, das Handeln und die Lebensbedingungen der Menschen in den Industrie- und Schwellenländern. Dieser Einfluss wirkt auf allen Ebenen der Gesellschaft bis auf das unternehmerische Verhalten mittelständischer Firmen. Daher wird anhand eines methodischen Rahmens versucht, aus der Sicht derer, die Naturwissenschaft, Technik und Wirtschaft für sich nutzbringend vorantreiben und aus der Sicht jener, die ohne Nutzen nur Betroffene sind, die wesentlichen Ziele, Kriterien und möglichen Folgen der daraus entstehenden Prozesse zu ermitteln, zu hinterfragen und nach festzulegenden Kriterien zu bewerten.		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester
Lehrveranstaltungen		
Technikfolgeabschätzung / -bewertung Lehrformen: Vorlesung und Übung Dozenten: Dr. Ulrich Jentzsch Sprache: Deutsch		2,00 SWS
Inhalte: Vermittlung von Grundkenntnissen: <ul style="list-style-type: none"> • zu den Begriffsinhalten einer TFA/TFB • zu zentralen Themenfeldern und Fragestellungen einer TFA/TFB • zu den Möglichkeiten und Grenzen prognostischer Aussagen im Rahmen einer TFA/TFB • zum prinzipiellen inhaltlichen Aufbau und einer formalen Struktur einer TFA/TFB • zu häufig verwendeten Methoden zur Problem- bzw. Entscheidungsanalyse innerhalb einer TFA/TFB - mit Übungen 		

Prüfung

schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten

Modul SME-Phy-B Physical Computing <i>Physical Computing</i>		6 ECTS / 180 h
Version 1.0.0 (seit WS14/15) Modulverantwortliche/r: Prof. Dr. Diedrich Wolter		
<p>Inhalte:</p> <p>Die in der physikalischen Umwelt eingebetteten Systeme können durch Sensoren ihre Umgebung erfassen und Wissen über ihren jeweiligen Kontext erlangen. Ziel dieses Kurses ist es, einen Überblick über die Möglichkeiten und Herausforderungen von Anwendungen im Bereich Physical Computing zu vermitteln. Physical Computing ist ein neues Gebiet an der Schnittstelle zu intelligenter Interaktion, eingebetteten Systemen und Smart Environments.</p> <p>Dieses Modul setzt zwei Schwerpunkte: Erstens, Kennenlernen von eingebetteten Sensorsystemen sowie Sammeln praktischer Erfahrung mit deren Programmierung und, zweitens, Algorithmen zur Verarbeitung von Sensordaten mit der Zielsetzung, Handlungen und Ereignisse zu erkennen und zu klassifizieren.</p> <p>Im Rahmen des Kurses programmieren Studierende in Kleingruppen ein eingebettetes System mit Sensorik und untersuchen Algorithmen zur Interpretation der Sensordaten:</p> <ul style="list-style-type: none"> • Modellbasierte Interpretation unsicherer Information • Aktions- und Prozesserkennung mit Markov-Modellen • Sensorfusion mit dem Kalmanfilter 		
<p>Lernziele/Kompetenzen:</p> <ul style="list-style-type: none"> • Einblick in die Programmierung eingebetteter Systeme • Erfahrung in der hardwarenahen Programmierung sammeln • Übersicht über Sensoren erlangen • Eignung von Sensoren zur Erkennung von Kontext und Umweltprozessen beurteilen • Kennenlernen von Algorithmen zur Interpretation von Sensordaten 		
<p>Bemerkung:</p> <p>The main language of instruction in this course is German. Lectures and tutorials may be delivered in English on demand.</p>		
<p>Empfohlene Vorkenntnisse:</p> <p>Elementare Grundkenntnisse der Programmierung sind dringend empfohlen (z.B. Modul MI-AuD-B), Grundkenntnisse in der Programmiersprache C (z.B. Modul SWT-IPC-B) können hilfreich sein.</p>		<p>Besondere Bestehensvoraussetzungen:</p> <p>Siehe Prüfungsordnung</p>
<p>Angebotshäufigkeit: SS, jährlich</p>	<p>Empfohlenes Fachsemester:</p>	<p>Minimale Dauer des Moduls:</p> <p>1 Semester</p>

Lehrveranstaltungen	
<p>1. Vorlesung Physical Computing</p> <p>Lehrformen: Vorlesung</p> <p>Dozenten: Prof. Dr. Diedrich Wolter</p> <p>Sprache: Deutsch</p> <hr/> <p>Lernziele:</p> <p>siehe Modulbeschreibung</p> <hr/> <p>Inhalte:</p>	2,00 SWS

siehe Modulbeschreibung	
Literatur: wird in der ersten Vorlesung bekanntgegeben	
2. Übung zu Physical Computing Lehrformen: Übung Dozenten: Prof. Dr. Diedrich Wolter Sprache: Deutsch	2,00 SWS
Inhalte: Praktische Übungen zu den Inhalten der Vorlesung	
Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten	

Modul Stat-B-01 Methoden der Statistik I		6 ECTS / 180 h
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Susanne Rässler		
Inhalte: Im Rahmen der Vorlesung Methoden der Statistik I und der zugehörigen Übung werden die wichtigsten Grundlagen und Methoden der deskriptiven (beschreibenden) Statistik vermittelt. Dabei umfasst der Abschnitt zur deskriptiven Statistik Methoden, mit denen ein gegebenes Datenmaterial überschaubar dargestellt bzw. durch wenige aussagekräftige Zahlen wie Lageparameter, Streuungsmaße oder Korrelationskoeffizienten charakterisiert werden kann. Schließlich werden verschiedene Fragen der Datenerhebung angesprochen, denn eine noch so ausgefeilte statistische Methode ist nur so gut, wie die Daten, auf die sie angewendet wird.		
Lernziele/Kompetenzen: keine		
Bemerkung: http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/		
Empfohlene Vorkenntnisse: keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, SS	Empfohlenes Fachsemester: ab dem 1.	Minimale Dauer des Moduls: 1 Semester Semester

Lehrveranstaltungen	
Methoden der Statistik I Lehrformen: Vorlesung und Übung Sprache: Deutsch	4,00 SWS
Lernziele: Die Studierenden sollen mit den grundlegenden statistischen Methoden vertraut gemacht werden. Besondere Schwerpunkte bilden dabei die theoretischen Grundlagen dieser Methoden, die Voraussetzungen ihrer Anwendbarkeit, ihre Umsetzung in Statistiksoftware sowie die sinnvolle Interpretation der Ergebnisse.	
Inhalte: Im Rahmen der Vorlesung Methoden der Statistik I und der zugehörigen Übung werden die wichtigsten Grundlagen und Methoden der deskriptiven (beschreibenden) Statistik vermittelt. Dabei umfasst der Abschnitt zur deskriptiven Statistik Methoden, mit denen ein gegebenes Datenmaterial überschaubar dargestellt bzw. durch wenige aussagekräftige Zahlen wie Lageparameter, Streuungsmaße oder Korrelationskoeffizienten charakterisiert werden kann. Schließlich werden verschiedene Fragen der Datenerhebung angesprochen, denn eine noch so ausgefeilte statistische Methode ist nur so gut, wie die Daten, auf die sie angewendet wird.	

Prüfung	
----------------	--

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

Zulassungsvoraussetzung zur Modulprüfung:

keine

Modul Stat-B-02 Methoden der Statistik II		6 ECTS / 180 h
Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Susanne Rässler		
<p>Inhalte:</p> <p>Im Rahmen der Vorlesung Methoden der Statistik II und der zugehörigen Übung werden die wichtigsten Grundlagen der Wahrscheinlichkeitsrechnung und der induktiven (schließenden) Statistik vermittelt. Im Einzelnen befasst sich die Vorlesung Methoden der Statistik II mit den grundlegenden Begriffen, Regeln und Gesetzmäßigkeiten der Wahrscheinlichkeitsrechnung, wobei vor allem Zufallsvorgänge, die sich durch sog. Zufallsvariablen beschreiben lassen, im Vordergrund des Interesses stehen. Viele aus der deskriptiven Statistik bekannte Größen, wie die Verteilungsparameter, können analog für Zufallsvariablen definiert werden. Außerdem werden mit dem Gesetz der großen Zahlen und dem zentralen Grenzwertsatz zwei für die induktive Statistik besonders wichtige Sätze der Wahrscheinlichkeitsrechnung vorgestellt.</p> <p>Des Weiteren stehen in der Vorlesung zur induktiven Statistik Methoden im Vordergrund, nach denen wahrscheinlichkeitstheoretisch fundierte Rückschlüsse von einer Stichprobe auf die betrachtete Grundgesamtheit möglich sind. Aufbauend auf den zuvor behandelten Grundlagen der Wahrscheinlichkeitstheorie werden Verfahren der Punktschätzung und der Intervallschätzung sowie wichtige Hypothesentests behandelt. Im Anschluss daran folgt ein Überblick über einige weitere interessante Teilgebiete der Statistik, wobei speziell die Methoden der Regressionsrechnung und der Analyse kategorialer Variablen ausführlicher besprochen werden.</p>		
<p>Lernziele/Kompetenzen:</p> <p>keine</p>		
<p>Bemerkung:</p> <p>http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/</p>		
<p>Empfohlene Vorkenntnisse:</p> <p>Die vorherige Absolvierung von Stat-B-01 (Methoden der Statistik I)</p>		<p>Besondere Bestehensvoraussetzungen:</p> <p>Siehe Prüfungsordnung</p>
<p>Angebotshäufigkeit: WS, SS</p>	<p>Empfohlenes Fachsemester:</p> <p>ab dem 2.</p>	<p>Minimale Dauer des Moduls:</p> <p>1 Semester</p>

Lehrveranstaltungen	
<p>Methoden der Statistik II Lehrformen: Vorlesung und Übung Sprache: Deutsch</p> <hr/> <p>Lernziele: Die Studierenden sollen mit den grundlegenden statistischen Methoden vertraut gemacht werden. Besondere Schwerpunkte bilden dabei die theoretischen Grundlagen dieser Methoden, die Voraussetzungen ihrer Anwendbarkeit, ihre Umsetzung in Statistiksoftware sowie die sinnvolle Interpretation der Ergebnisse.</p> <hr/> <p>Inhalte: Im Rahmen der Vorlesung Methoden der Statistik II und der zugehörigen Übung werden die wichtigsten Grundlagen der Wahrscheinlichkeitsrechnung und der induktiven (schließenden) Statistik vermittelt. Im Einzelnen befasst sich die Vorlesung Methoden der Statistik II mit den grundlegenden Begriffen, Regeln</p>	4,00 SWS

und Gesetzmäßigkeiten der Wahrscheinlichkeitsrechnung, wobei vor allem Zufallsvorgänge, die sich durch sog. Zufallsvariablen beschreiben lassen, im Vordergrund des Interesses stehen. Viele aus der deskriptiven Statistik bekannte Größen, wie die Verteilungsparameter, können analog für Zufallsvariablen definiert werden. Außerdem werden mit dem Gesetz der großen Zahlen und dem zentralen Grenzwertsatz zwei für die induktive Statistik besonders wichtige Sätze der Wahrscheinlichkeitsrechnung vorgestellt.

Des Weiteren stehen in der Vorlesung zur induktiven Statistik Methoden im Vordergrund, nach denen wahrscheinlichkeits-theoretisch fundierte Rückschlüsse von einer Stichprobe auf die betrachtete Grundgesamtheit möglich sind. Aufbauend auf den zuvor behandelten Grundlagen der Wahrscheinlichkeitstheorie werden Verfahren der Punktschätzung und der Intervallschätzung sowie wichtige Hypothesentests behandelt. Im Anschluss daran folgt ein Überblick über einige weitere interessante Teilgebiete der Statistik, wobei speziell die Methoden der Regressionsrechnung und der Analyse kategorialer Variablen ausführlicher besprochen werden.

Literatur:

Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G.: Statistik: Der Weg zur Datenanalyse, 5. Auflage, Springer, Heidelberg, 2004.

Agresti, A., Franklin, C. A.: Statistics: The Art and Science of Learning from Data, Prentice Hall, Upper Saddle River (New Jersey), 2006.

Krämer, W.: So lügt man mit Statistik, 8. Auflage, Piper, Frankfurt a.M., 2000.

Vogel, F.: Beschreibende und schließende Statistik - Formeln, Definitionen, Erläuterungen, Stichwörter und Tabellen, 13. Auflage, München, 2005.

Vogel, F.: Beschreibende und schließende Statistik - Aufgaben und Beispiele, 9. Aufl., München, 2001.

Prüfung

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

Zulassungsvoraussetzung zur Modulprüfung:

keine

Modul SWT-FSA-B Foundations of Software Analysis <i>Foundations of Software Analysis</i>		6 ECTS / 180 h
Version 2.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: This module introduces students to the mathematical and practical foundations of software analysis, which are at the heart of modern techniques for software verification and compiler optimization and key for enhancing software quality.		
Lernziele/Kompetenzen: On completion of this module, students will be able to understand key concepts, techniques and algorithms for software analysis and appreciate the workings of modern software analysis tools.		
Bemerkung: The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German. The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 30 hrs. attending lectures (Vorlesungen) • 30 hrs. attending practicals (Übungen) • 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources • 60 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium) 		
Empfohlene Vorkenntnisse: Basic knowledge in discrete mathematics and logics, such as acquired in the module "Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik)" (GdI-MfI-1).		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Foundations of Software Analysis Lehrformen: Vorlesung Dozenten: Prof. Dr. Gerald Lüttgen Sprache: Englisch/Deutsch	2,00 SWS
Inhalte: Students will be introduced to the foundations of software analysis and their applications to software verification and code analysis and optimization. Particular emphasis will be put on semantics and abstraction, and their underlying mathematical theories based on lattices and order. The following topics will be covered: (i) semantics of programs; (ii) abstraction and abstract interpretation; (iii) elementary fixed point theory; (iv) operational and denotational abstract semantics; (v) software verification based on the methods of Floyd and Hoare; (vi) code analysis and optimization based on data flow analysis; (vii) outlook on advanced, modern aspects of software analysis.	

<p>Literatur:</p> <ul style="list-style-type: none"> • Berghammer, R. Semantik von Programmiersprachen, Berlin, Logos Verlag 2001. • Nielson, H. R., Nielson, F., Semantics with Applications: An Appetizer, Springer, 2007. • Nielson, F., Nielson, H. R., Hankin, C. Principles of Program Analysis, Springer, 1999. • Loeckx, J. and Sieber, K. The Foundations of Program Verification, 2nd ed. Wiley, 1987. • Davey, B. A., Priestley, H. A. Introduction to Lattices and Order, 2nd ed. Cambridge University Press, 2002. 	
<p>2. Übung Foundations of Software Analysis</p> <p>Lehrformen: Übung</p> <p>Dozenten: Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen</p> <p>Sprache: Englisch/Deutsch</p> <hr/> <p>Inhalte:</p> <p>The practicals (Übungen) will deepen the concepts and techniques taught in the lectures (Vorlesungen), and apply them to the analysis of small examples of software. They will mainly cover pen-and-paper exercises, but will also introduce students to modern software analysis tools. Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students.</p> <hr/> <p>Literatur:</p> <p>- see the corresponding lectures -</p>	2,00 SWS
<p>Prüfung</p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten</p> <p>Bearbeitungsfrist: 3 Wochen</p> <p>Beschreibung:</p> <p>Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen).</p> <p>Colloquium (Kolloquium) consisting of questions testing the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen), on the basis of the submitted solutions to the assignment (Hausarbeit).</p>	

Modul SWT-FSE-B Foundations of Software Engineering		6 ECTS / 180 h
<i>Foundations of Software Engineering</i>		
Version 1.0.0 (seit SS14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: This module teaches the foundations of software engineering that are applicable to various kinds of software systems – from information systems to embedded systems. It focusses on technologies, notations and processes for system specification, design, implementation, and verification and validation.		
Lernziele/Kompetenzen: Students will receive an introduction to the common problems and paradigms in, and foundations of, software development. They will also gather conceptual and practical knowledge in the analysis, design and testing of software, with an emphasis on technical aspects of specifying, designing, implementing, verifying and validating software.		
Bemerkung: The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German. The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 30 hrs. attending lectures (Vorlesungen) • 45 hrs. reviewing the lectures, including researching and studying material from additional sources • 30 hrs. attending practicals (Übungen) • 45 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources • 30 hrs. preparing for the written exam (Klausur) 		
Empfohlene Vorkenntnisse: Basic knowledge in Computer Science, as well as knowledge in programming in Java and in algorithms and data structures.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Foundations of Software Engineering Lehrformen: Vorlesung Dozenten: Prof. Dr. Gerald Lüttgen Sprache: Englisch/Deutsch	2,00 SWS 0 ECTS
Inhalte: The lectures (Vorlesungen) provide an introduction to the foundations of software engineering, including commonly used technologies, notations and processes for all software engineering phases. In particular, conceptual and technical aspects of software specification, architecture and design, and verification and validation are discussed, such as the Unified Modeling Language (UML) and its semantics,	

<p>model-driven and pattern-based development, and software testing. Students are also introduced to specific aspects of agile software development.</p> <p>Literatur:</p> <ul style="list-style-type: none"> • Sommerville, I. Software Engineering, 9th ed. Addison-Wesley, 2010. • Robertson, S. and Robertson, J. Mastering the Requirements Process, 2nd ed. Addison-Wesley, 2006. • Stevens, P. and Pooley, R. Using UML - Software Engineering with Objects and Components, 2nd. ed. Addison-Wesley, 2008. • Freeman, E., Freeman, E., Sierra, K. and Bates, B. Head First Design Patterns. O'Reilly, 2004. • Gamma, E., Helm, R., Johnson, R. and Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Design. Addison-Wesley, 1994. <p>Further literature will be announced in the lectures.</p>	
<p>2. Übung Foundations of Software Engineering</p> <p>Lehrformen: Übung</p> <p>Dozenten: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen</p> <p>Sprache: Englisch/Deutsch</p> <hr/> <p>Inhalte:</p> <p>The practicals (Übungen) exercise and deepen the conceptual knowledge transferred via the lectures (Vorlesungen), and relay practical knowledge in software engineering.</p> <hr/> <p>Literatur:</p> <p>- see the corresponding lectures -</p>	<p>2,00 SWS 0 ECTS</p>
<p>Prüfung</p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p>Beschreibung:</p> <p>Written exam (Klausur) consisting of questions that relate to the contents of the lectures (Vorlesungen) and practicals (Übungen) of this module. The written exam is passed if at least 50% of the available points are reached.</p>	

Modul SWT-IPC-B Imperative Programming Using C <i>Imperative Programming Using C</i>		3 ECTS / 90 h
Version 3.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: The module covers the basic syntax of the C programming language, including types, operations and control structures. Concepts such as pointers, memory management, I/O handling and POSIX threads will be discussed in detail. Furthermore, it will be explained how the compiler, pre-processor, debugger, "make" tool and external libraries are employed.		
Lernziele/Kompetenzen: Students will develop an in-depth understanding of the C programming language, and acquire practical programming skills by learning how to develop clearly written and well-structured programs in ANSI C.		
Bemerkung: The main language of instruction is English. The practicals may be delivered in German if all participating students are fluent in German. The total workload of 90 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 30 hrs. attending practicals (Übungen) • 15 hrs. preparing and reviewing the practicals (including solving exercises in private study) • 45 hrs. working on the written assignment (Hausarbeit) and preparing for the colloquium (Kolloquium) 		
Empfohlene Vorkenntnisse: Basic knowledge in programming and in algorithms and data structures. Additionally, elementary knowledge in computer architectures and operating systems is desirable.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Übung Imperative Programming Using C Lehrformen: Übung Dozenten: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen Sprache: Englisch/Deutsch	2,00 SWS
Inhalte: The practicals teach various topics of the C programming language, as mentioned under section "Modulinhalte" above. In addition, they interleave this knowledge transfer with numerous practical examples and small programming tasks.	
Literatur: <ul style="list-style-type: none"> • Kernighan, B. W. and Ritchie, D. The C Programming Language, 2nd ed. Prentice Hall, 1988. 	
Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten	

Bearbeitungsfrist: 3 Wochen

Beschreibung:

Assignment (Hausarbeit) involving the production and documentation of software in the C programming language, which has been developed during the practicals (Übungen).

Colloquium (Kolloquium) consisting of questions concerning the C programming language, and critical discussion of the documented software in the assignment (Hausarbeit).

Modul SWT-PCC-M Principles of Compiler Construction		6 ECTS / 180 h
<i>Principles of Compiler Construction</i>		
Version 3.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: The module teaches the theoretical and practical principles of compiler construction, from lexical analysis and parsing, to semantic analysis, to code generation and optimisation.		
Lernziele/Kompetenzen: On completion of this module, students will be familiar with all phases of a modern compiler – from lexical analysis and parsing, to semantic analysis and finally code generation and code optimisation – and will have a deep understanding of the workings of compilers. As a result, students will be able to use compilers more effectively and learn better debugging practices. Students will also be able to start building compilers on their own.		
Bemerkung: The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German. The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 30 hrs. attending lectures (Vorlesungen) • 30 hrs. reviewing the lectures, including researching and studying material from additional sources • 30 hrs. attending practicals (Übungen) • 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources • 60 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium) 		
Empfohlene Vorkenntnisse: Basic knowledge in the theoretical foundations of Computer Science (especially in language theory and automata theory) and in algorithms and data structures.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
1. Vorlesung Principles of Compiler Construction Lehrformen: Vorlesung Dozenten: Prof. Dr. Gerald Lüttgen Sprache: Englisch/Deutsch	2,00 SWS
Inhalte: Students will be familiarised with a variety of theoretical and practical concepts, techniques and algorithms employed in compiler construction, which reach from language theory, to automata theory, to data flow analysis. The lectures will focus on the following aspects of compiler construction: lexical analysis, parsing, abstract syntax, semantic analysis, code generation and code optimisation.	
Literatur:	

<ul style="list-style-type: none"> • Louden, K. C. Compiler Construction: Principles and Practice. Course Technology, 1997. • Aho, A. V., Sethi, R., Ullman, J. D. and Lam, M. S. Compilers: Principles, Techniques, and Tools, 2nd ed. Addison-Wesley, 2006. • Fischer, C. N., Cytron, R. K. and LeBlanc Jr., R. J. Crafting a Compiler. Pearson, 2010. • Muchnick, S. S. Advanced Compiler Design and Implementation, Morgan Kaufmann, 1997. 	
<p>2. Übung Principles of Compiler Construction</p> <p>Lehrformen: Übung</p> <p>Dozenten: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen</p> <p>Sprache: Englisch/Deutsch</p> <hr/> <p>Inhalte:</p> <p>Students will practice the theoretical concepts taught in the lectures by applying them to a variety of exercises, so that they can appreciate the diverse range of foundations that make modern programming languages possible. The exercises will largely be pen-and-paper exercises but may also involve some work using computers. Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students, within the timetabled practicals (Übungen). Students can gain further practical experience in compiler construction by simultaneously attending the module "Bachelorprojekt Softwaretechnik und Programmiersprachen" (SWT-PR1-B).</p> <hr/> <p>Literatur:</p> <p>- see the corresponding lectures -</p>	2,00 SWS
<p>Prüfung</p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten</p> <p>Bearbeitungsfrist: 3 Wochen</p> <p>Beschreibung:</p> <p>Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures (Vorlesungen) and practicals (Übungen).</p> <p>Colloquium (Kolloquium) consisting of questions testing the knowledge transferred in the lectures (Vorlesungen) and practicals (Übungen), on the basis of the submitted solutions to the assignment (Hausarbeit).</p>	

<p>Modul SWT-PR2-B SWT Bachelorprojekt Software Systems Science <i>SWT Bachelors Project in Software Systems Science</i></p>	<p>12 ECTS / 360 h</p>	
<p>Version 1.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen</p>		
<p>Inhalte: Ein überschaubares Projekt zu Themen aus der aktuellen Forschungsarbeit auf den Gebieten Modellierung, Programmierung, Entwicklung, Analyse und Verifikation von Softwaresystemen wird in zum Teil gemeinsam, zum Teil arbeitsteilig arbeitenden Kleingruppen von Studierenden – und ggf. auch einzeln – von der Konzeption bis zur Umsetzung und deren Dokumentation und Präsentation durchgeführt. Dabei geht es insbesondere um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe. Dies erfordert eine umfangreiche Recherche aktueller Literatur und zum Thema vorgeschlagener Ansätze, sowie eine detaillierte Planung, deren Ergebnisse in einem Planungsworkshop vorgestellt und diskutiert werden. Beispiele für mögliche Projektthemen sind: Modellierung und Verifikation einer Robotersteuerung, Implementierung und Analyse von Algorithmen zur Programmverifikation, Entwicklung und Validierung einer Anwendersoftware für Mehrkernrechner.</p>		
<p>Lernziele/Kompetenzen: Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Softwaresystem-Projekten auftretenden konzeptionellen und praktischen Probleme wie auch von erfolgsversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich von Softwaretechnik und Programmiersprachen in Kleingruppen – oder ggf. auch einzeln – geschieht, gewinnen die Studierenden wichtige Erfahrungen in der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in wissenschaftlich ausgerichteten Arbeitsberichten und in der Präsentation dieser Ergebnisse.</p>		
<p>Bemerkung: Dieses Modul erstreckt sich über 2 Semester: 2x6=12 ECTS, 2x4=8 SWS. Der Arbeitsaufwand beträgt insgesamt 360 Std., welche sich grob wie folgt gliedern:</p> <ul style="list-style-type: none"> • 60 Std. Recherche, Planung und Teilnahme am Planungsworkshop • 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien • 180 Std. Durchführung des Projekts (Projektarbeit) • 20 Std. Erstellung des Zwischenberichts (Hausarbeit) • 60 Std. Erstellung des Abschlussberichts, sowie Erstellung und Präsentation des Projektposters (Hausarbeit und Kolloquium) <p>Berichte und Posterpräsentation dürfen wahlweise in Deutsch oder Englisch abgefasst sein.</p>		
<p>Empfohlene Vorkenntnisse: Grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z. B. erworben im Modul "Software Engineering Lab" (SWT-SWL-B), und zum wissenschaftlichen Arbeiten, z. B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik" (IAI-WAI-B).</p>	<p>Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung</p>	
<p>Angebotshäufigkeit: SS, jährlich</p>	<p>Empfohlenes Fachsemester:</p>	<p>Minimale Dauer des Moduls:</p>

		2 Semester
Lehrveranstaltungen		
Übung SWT Bachelorprojekt Software Systems Science Lehrformen: Übung Dozenten: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen Sprache: Englisch/Deutsch		8,00 SWS
Lernziele: Werden zu Beginn des Projekts bekannt gegeben.		
Inhalte: Durchführung des Projekts. Begleitend werden Tutorien angeboten, innerhalb derer die Studierenden mit themenrelevanten Methoden, Techniken und Softwarewerkzeugen vertraut gemacht werden. Darüber hinaus finden regelmäßige Treffen zwischen Teilnehmerinnen/Teilnehmern und Projektbetreuern statt, indem der Projektfortschritt kontrolliert und fachlich diskutiert wird. Der Fortschritt bzw. die Ergebnisse des Projekts werden von Studierenden in einem Zwischen- und Abschlussbericht dokumentiert. Zudem wird von jeder Kleingruppe – bzw. bei Individualprojekten von der Teilnehmerin bzw. dem Teilnehmer – ein Poster zum Projekt und dessen Ergebnissen erstellt, welches im Rahmen einer Abschlussveranstaltung präsentiert, erläutert und diskutiert wird.		
Literatur: Je nach Projektthematik; wird zu Beginn des Projekts bekannt gegeben.		
Prüfung schriftliche Hausarbeit / Bearbeitungsfrist: 12 Wochen Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung. Beschreibung: Anfertigen eines schriftlichen Zwischenberichts zum Projekt nach etwa 80 Std. geleisteter Projektarbeit, spätestens am Ende des Semesters, in dem das Projekt begonnen wurde.		
Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 12 Wochen Zulassungsvoraussetzung zur Modulprüfung: Regelmäßige Teilnahme an der Lehrveranstaltung. Beschreibung: Kolloquium: Fachliche Diskussion auf der Grundlage des im Projekt bearbeiteten Themas im Rahmen einer Abschlussveranstaltung, auf der zunächst das		

zum Projekt angefertigte Poster erläutert wird; hier können auch praktische Projektergebnisse (z. B. lauffähige Software) demonstriert werden.

Hausarbeit: Anfertigen eines schriftlichen Abschlussberichts nach abgeschlossener Projektarbeit.

Modul SWT-SSP-B Soft Skills in IT-Projekten <i>Soft Skills for IT Projects</i>		3 ECTS / 90 h
Version 1.0.0 (seit SS14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: Ziel des Moduls ist es, die in der Praxis der IT-Projekte immer wichtiger werdenden Soft Skills wissenschaftlich und methodisch fundiert zu vermitteln. Die Studierenden lernen, dieses Wissen in der Praxis ziel- und lösungsorientiert anwenden zu können.		
Lernziele/Kompetenzen: Nach erfolgreicher Teilnahme an diesem Modul können Studierende insbesondere: <ul style="list-style-type: none"> • Sich die Bedeutung menschlicher Faktoren in großen IT-Projekten bewusst machen; • Erfolgsfaktoren der Teamarbeit kennen und einschätzen; • Eigenkompetenzen und Kompetenzen anderer wahrnehmen, beurteilen und für die Teamorganisation nutzen; • Muster der Gruppendynamik - insbes. Kommunikationsmuster, Konfliktsituationen und Verantwortungsdiffusion - erkennen und managen. 		
Bemerkung: Der Arbeitsaufwand beträgt 90 Std., welche sich grob wie folgt gliedern: <ul style="list-style-type: none"> • 30 Std. Teilnahme an der Vorlesung und Übung • 45 Std. Vor- und Nachbereitung der Vorlesung und Übung • 15 Std. Vorbereitung auf die Klausur 		
Empfohlene Vorkenntnisse: Keine		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: SS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Soft Skills in IT-Projekten Lehrformen: Vorlesung und Übung Dozenten: Norbert Seifert Sprache: Deutsch	2,00 SWS 3 ECTS
Lernziele: Die Lernziele der Lehrveranstaltung decken sich mit denen des Moduls.	
Inhalte: Der Inhalt orientiert sich an der in der Praxis großer IT-Projekte erforderlicher Soft Skills: <ol style="list-style-type: none"> 1. Vorsprung durch Menschenkenntnis; 2. Teamorganisation und -aufstellung; 3. Kommunikation und Konfliktmanagement; 4. Motivationsfaktoren und Selbstverantwortung; 5. Menschliche Spielregeln großer IT-Projekte. 	

Literatur: Literatur wird zu Beginn der Lehrveranstaltung angegeben.	
Prüfung schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten Beschreibung: Die Klausur prüft Wissen und Verständnis der in der Vorlesung und Übung vermittelten Lehrinhalte.	

Modul SWT-SWL-B Software Engineering Lab <i>Software Engineering Lab</i>		6 ECTS / 180 h
Version 3.0.0 (seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
Inhalte: Small teams of students will conduct a software project, starting from a brief problem description. This involves the application of modern software engineering tools, skills in collaboration and team organisation, and knowledge of processes and techniques for producing software artefacts and associated documents.		
Lernziele/Kompetenzen: Students will develop a piece of medium-sized software in small teams, thereby acquiring practical expertise in software engineering and skills in working in a software development team. In addition, this module deepens the students' programming proficiency and their understanding of flexible software engineering processes and of software and process quality, and familiarises them with the deployment and use of modern software engineering tools.		
Bemerkung: The main language of instruction is English. The practicals may be delivered in German if all participating students are fluent in German. The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> • 15 hrs. attending meetings of the student's team with the lecturer (Dozent) on planning, coordination and feedback • 10 hrs. attending the accompanying practicals/tutorials (Übungen/Tutorials) on software tools • 130 hrs. conducting the team project • 25 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium) 		
Empfohlene Vorkenntnisse: Basic knowledge in Computer Science and Software Engineering, as well as knowledge in Java programming and in programming in the small.		Besondere Bestehensvoraussetzungen: Siehe Prüfungsordnung
Angebotshäufigkeit: WS, jährlich	Empfohlenes Fachsemester:	Minimale Dauer des Moduls: 1 Semester

Lehrveranstaltungen	
Übung Software Engineering Lab Lehrformen: Übung Dozenten: Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen Sprache: Deutsch/Englisch	4,00 SWS
Inhalte: Each team will carry out a software project. It will also regularly meet with their tutor (Dozent) in order to critically reflect on the team's work, and participate in tutorials that introduce the software engineering tools and some software engineering techniques to be used in this project.	
Literatur:	

<ul style="list-style-type: none"> • Tachiev, P., Leme, F., Massol, V. and Gregory, G. JUnit in Action, 2nd ed. Manning Publications, 2010. • Loeliger, J. and McCullough, M. Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development, 2nd ed. O'Reilly, 2012. • Vogel, L. Eclipse IDE. Lars Vogel, 2013. ISBN 3943747042. • Schwaber, K. and Beedle, M. Agile Software Development with Scrum, Prentice Hall, 2001 • Cohn, M. User Stories Applied. Addison-Wesley, 2004. <p>See the description of the module "Foundations of Software Engineering (SWT-FSE-B)" for further literature.</p>	
<p>Prüfung Hausarbeit mit Kolloquium / Prüfungsdauer: 45 Minuten Bearbeitungsfrist: 2 Wochen</p> <p>Beschreibung: Assignment (Hausarbeit) involving the compilation of a written project report by each team, which shall cover the following topics:</p> <ul style="list-style-type: none"> • A description of the team's produced artefacts, including the electronic submission of the artefacts themselves; • A description, justification and critical reflection of the employed software engineering processes, methods and techniques in general and in each development phase; • A description of the team's organisation, the distribution of work and the contributions of each team member. <p>The submission deadline and the details of the required content and format of this report will be announced at the beginning of the semester.</p> <p>Colloquium (Kolloquium) consisting of a critical discussion of the team's produced software and project report with respect to the taken design decisions and possible alternatives, the quality of the produced artefacts and documentation, the project's status and completeness, the conduct of testing, and the appropriateness of the employed techniques and processes. The colloquium takes place in the presence of the team as a whole, but each question will be addressed to a specific student so that marks can be individualised.</p> <p>Because this module involves a team effort, the examination can only be resit in a winter semester. In addition, this module calls for active participation throughout.</p>	