Otto-Friedrich Universität Bamberg

# Modulhandbuch

## Bachelor Informatik: Software Systems Science

## Wirtschaftsinformatik und Angewandte Informatik

**Gemäß der geltenden Fassung der Studien- und Fachprüfungsordnung vom 28.09.2018 für den Bachelorstudiengang Informatik: Software Systems Science an der Otto-Friedrich-Universität Bamberg. Gültig ab Wintersemester 2024/25.**

**Stand: 20.09.2024**

**Hinweis zur Weitergeltung älterer Fassungen eines Modulhandbuchs:**

## 1. Geltungsbeginn

Die im vorliegenden Modulhandbuch enthaltenen Modulbeschreibungen gelten erstmals für das Semester, das auf dem Deckblatt angegeben ist.

## 2. Übergangsbestimmung

1.
Mit der ab 1. Oktober 2022 geltenden Änderung der Studien- und Fachprüfungsordnung werden die ECTS-Grenzen des Pflichtbereichs A1 und des Wahlpflichtbereichs A4 geändert. Die Veränderungen der ECTS-Grenzen des Pflichtbereichs A1 und des Wahlpflichtbereichs A5 gelten nicht für Studierende, die das Modul WiMa-B-01b vor Inkrafttreten dieser Änderungssatzung bereits ganz oder in Teilen absolviert haben.
Betroffene Studierende absolvieren im Pflichtbereich A1 28 ECTS und im Wahlpflichtbereich A 4 10-18 ECTS, sofern in den Wahlpflichtbereichen gemäß Anhang 1 Satz 4 35-41 ECTS absolviert werden.

2.
Im Übrigen gilt hinsichtlich des geltenden Modulhandbuchs Folgendes:

a. Studierende, die gemäß bisher geltendem Modulhandbuch ein Modul bereits in Teilen absolviert haben (vgl. Nr. 2b), schließen das Modul nach der bisher geltenden Fassung des Modulhandbuchs ab.

   Diese Übergangsbestimmung gilt ausschließlich für den dem versäumten/nicht bestandenen/nicht absolvierten regulären Prüfungstermin unmittelbar folgenden Prüfungstermin. Auf Antrag der oder des Studierenden kann der Prüfungsausschuss in begründeten Fällen eine Verlängerung der Übergangsfrist festlegen.

b. Ein Modul ist in Teilen absolviert, wenn die Modulprüfung nicht bestanden oder versäumt wurde. Gleiches gilt für den Fall, dass zumindest eine Modulteilprüfung bestanden, nicht bestanden oder versäumt wurde.

   Ferner gilt ein Modul als in Teilen absolviert, sofern sich die oder der Studierende gemäß bisher geltendem Modulhandbuch zu einer dem jeweiligen Modul zugeordneten Lehrveranstaltung angemeldet hat.

## 3. Geltungsdauer

Das Modulhandbuch gilt bis zur Bekanntgabe eines geänderten Modulhandbuchs auch für nachfolgende Semester.

## Abweichungen im Modulangebot des BSc Informatik: Software System Science gegenüber der StuFPO vom 28.09.2018, zuletzt geändert am 28.02.2024

Modulgruppen A1 & A2

- Das Modul KTR-MfI-2 wird nicht mehr angeboten. Es wird durch das Modul WiMa-B-001 ersetzt.
- Die Lehrveranstaltungen zu folgenden Modulen werden nicht mehr angeboten. Um das Modul zu absolvieren, muss jeweils auf das Lehrangebot zu einem anderen Modul zurückgegriffen und die zugehörige Lehrveranstaltung anteilig besucht werden:
    - A1: GdI-MfI-1: anteiliger Besuch von Inf-LBR-B
    - A2: DSG-EiAPS-B: anteiliger Besuch von Inf-Einf-B
    - A2: GdI-GTI-B: anteiliger Besuch von Inf-DM-B
    - A2: PSI-EiRBS-B: anteiliger Besuch von Inf-GRABS-B
- Die Prüfungsform zu PSI-IntroSP-B hat sich geändert: Die Zulassung zur Modulprüfung zu setzt voraus, dass die Studienleistung in Form eines als E-Prüfung durchgeführten Testats erfolgreich absolviert wurde.

Modulgruppe A3

- Das Modul GdI-IFP-B wird künftig als Mastermodul angeboten und ist daher nicht mehr im Bachelorstudiengang absolvierbar. Es wird ersetzt durch das Modul GdI-MTL-B. Dieses Modul entfällt daher im Wahlpflichtbereich der Modulgruppe A4.

Modulgruppe A6

- Die Prüfungsform der Seminarmodule hat sich geändert: Sie lautet nun „Referat mit schriftlicher Hausarbeit oder schriftliche Hausarbeit mit Kolloquium“

Modulgruppe A8

- Die Prüfungsform der Bachelorarbeit hat sich geändert: Sie lautet nun „schriftliche Hausarbeit mit Kolloquium“

# Äquivalenzliste BA Informatik: Software Systems Science, StuFPO 28.09.2018

Im Folgenden finden Sie eine Auflistung von Modulen, deren Bezeichnung bzw. Kürzel geändert wurde, ohne dass damit eine wesentliche Änderung des Moduls verbunden ist. Sofern ein in der Spalte „bisheriges Modul“ aufgeführtes Modul erfolgreich absolviert wurde, kann das in der Spalte „neues Modul“ angegebene Modul nicht belegt werden.

| bisheriges Modul | | | neues Modul | | |
|---|---|---|---|---|---|
| **Modulkürzel** | **Modulbezeichnung** | **bis (Semester)** | **Modulkürzel** | **Modulbezeichnung** | **ab (Semester)** |
| DSG-IDistrSys | Einführung in Verteilte Systeme | WS18/19 | DSG-IDistrSys-B | Einführung in Verteilte Systeme | SS19 |
| SEDA-PT-B | Methoden der Präsentation, Gesprächsführung und Diskussion | WS18/19 | ISM-PT-B | Methoden der Präsentation, Gesprächsführung und Diskussion | SS19 |
| SEDA-TA-B | Technikfolgeabschätzung / -bewertung | WS18/19 | ISM-TA-B | Technikfolgeabschätzung / -bewertung | SS19 |
| GdI-IFP | Introduction to Functional Programming | SS19 | GdI-IFP-B | Introduction to Functional Programming | WS19/20 |
| MI-AuD-B | Algorithmen und Datenstrukturen | WS19/20 | AI-AuD-B | Algorithmen und Datenstrukturen | SS20 |
| SWT-PR1-B | Bachelorprojekt Softwaretechnik und Programmiersprachen | WS21/22 | SWT-SWP-B | Software Engineering Project | SS22 |
| SWT-FSA-B | Foundations of Software Analysis | SS22 | SWT-FPS-B | Foundations of Program Semantics | WS22/23 |
| AI-KI-B | Einführung in die Künstliche Intelligenz | WS23/24 | KogSys-KI-B | Einführung in die Künstliche Intelligenz | SS24 |
| xAI-MML-M | Mathematics for Machine Learning | SS24 | xAI-MML-B | Mathematics for Machine Learning | WS24/25 |
| AISE-Sem-B | Bachelorseminar Computational Philosophy | SS24 | AISE-SemCP-B | Bachelorseminar Computational Philosophy | WS24/25 |
| EESYS-IITP-B | Internationales IT-Projektmanagement | SS24 | AIC-IITP-B | Internationales IT-Projektmanagement | WS24/25 |

Stand der Äquivalenzliste: 17.07.2024

# Module

AI-AuD-B: Algorithmen und Datenstrukturen.....11

AIC-IITP-B: Internationales IT Projektmanagement.....14

AISE-SemCP-B: Bachelorseminar Computational Philosophy.....16

AISE-SemSC-B: Bachelorseminar Smart City.....18

AlgoK-AK-B: Algorithmen und Komplexität.....20

AlgoK-Sem-B: Bachelorseminar Algorithmen und Komplexitätstheorie.....23

AlgoK-TAG: Tree decompositions, algorithms and games.....24

CG-SemCGA-B: Seminar Computergrafik und Animation.....26

CG-SemVRAR-B: Seminar Virtual Reality / Augmented Reality.....27

DS-Sem-B: Bachelorseminar Dialogsysteme.....28

DSG-AJP-B: Fortgeschrittene Java Programmierung.....30

DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software.....32

DSG-IDistrSys-B: Introduction to Distributed Systems.....35

DSG-JaP-B: Java Programmierung.....38

DSG-PKS-B: Programmierung komplexer interagierender Systeme.....40

DSG-Project-B: Bachelorprojekt zur Praktischen Informatik.....42

DSG-Sem-B: Bachelorseminar zur Praktischen Informatik.....44

DT-CPP-B: Einführung in die Systemprogrammierung in C++.....46

DT-DB4MLKD-B: Moderne Datenbanksysteme für maschinelles Lernen und Wissensentdeckung.....47

DT-Proj-B: Bachelor-Projekt: Data Engineering.....48

GdI-GTI-B: Grundlagen der Theoretischen Informatik.....49

GdI-MTL-B: Modal and Temporal Logic.....51

GdI-MfI-1: Mathematik für Informatik 1 (Aussagen- u. Prädikatenlogik).....53

GdI-Proj-B: Bachelorprojekt Grundlagen der Informatik.....55

GdI-Sem-B: Bachelorseminar Grundlagen der Informatik.....57

HCI-IS-B: Interaktive Systeme.....59

HCI-Sem-B: Bachelorseminar Mensch-Computer-Interaktion.....62

HCI-US-B: Ubiquitäre Systeme.....64

Inf-DM-B: Diskrete Modellierung.....67

Inf-Einf-B: Einführung in die Informatik....69
Inf-GRABS-B: Grundlagen der Rechnerarchitektur und Betriebssysteme....72
Inf-LBR-B: Logik und Berechenbarkeit....74
KInf-Seminar-B: Bachelorseminar Kulturinformatik....78
KTR-Datkomm-B: Datenkommunikation....80
KTR-MfI-2: Mathematik für Informatik 2 (Lineare Algebra)....84
KTR-Proj: Projekt Kommunikationsnetze und -dienste....86
KTR-Sem-B: Bachelorseminar zu Kommunikationssystemen und Rechnernetzen....89
KogSys-GAI-B: Genderaspekte in der Informatik....91
KogSys-KI-B: Einführung in die Künstliche Intelligenz....93
KogSys-ML-B: Einführung in Maschinelles Lernen....96
KogSys-Sem-B: Bachelorseminar Kognitive Systeme....99
MI-Sem-B: Bachelorseminar zur Medieninformatik....101
MI-WAIAI-B: Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik....103
MI-WebT-B: Web-Technologien....105
MII-SemHRI-B: Bachelorseminar Mensch-Roboter-Interaktion....108
MOBI-DBS-B: Datenbanksysteme....109
MOBI-Proj-B: Bachelor Project Mobile Software Systems....111
MOBI-SEM-B: Bachelor-Seminar Mobile Software Systems....113
PSI-DatSchu-B: Datenschutz....114
PSI-EDS-B: Ethics for the Digital Society....115
PSI-EiRBS-B: Einführung in Rechner- und Betriebssysteme....117
PSI-IntroSP-B: Introduction to Security and Privacy....119
PSI-ProjectPAD: Project Practical Attacks and Defenses....122
PSI-Sem-B: Seminar Security and Privacy Foundations....125
SWT-FPS-B: Foundations of Program Semantics....127
SWT-FSE-B: Foundations of Software Engineering....130
SWT-SEM-B: Seminar Software Engineering and Programming Languages (Bachelor)....132
SWT-SWL-B: Software Engineering Lab....134
SWT-SWP-B: Software Engineering Project....136
SYSNAP-Project-B: Projekt Systemnahe Programmierung....138

SYSNAP-SEM-B: Seminar System Software ..... 140

SYSNAP-SNAP-B: Systemnahe Programmierung ..... 142

SoSySc-Thesis-B: Bachelorarbeit in Software Systems Science ..... 144

Stat-B-01: Methoden der Statistik I ..... 145

Stat-B-02: Methoden der Statistik II ..... 147

UxD-Sem-B: Bachelorseminar User Experience and Design ..... 149

VIS-GIV-B: Grundlagen der Informationsvisualisierung ..... 151

VIS-Sem-B: Bachelorseminar Informationsvisualisierung ..... 153

WI-Seminar-B: Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik ..... 155

WiMa-B-002: Wirtschaftsmathematik: Analysis ..... 156

xAI-MML-B: Mathematics for Machine Learning ..... 158

xAI-Sem-B1: Bachelorseminar Erklärbares Maschinelles Lernen ..... 160

# Übersicht nach Modulgruppen

## 1) A1 Fachstudium Mathematische Grundlagen (Modulgruppe) ECTS: 30

Studierende, die das Modul Wirtschaftsmathematik mit 4 ECTS absolviert haben (WiMa-B-01b), erreichen in dieser Modulgruppe **28 ECTS**.

GdI-MfI-1: Mathematik für Informatik 1 (Aussagen- u. Prädikatenlogik) (6 ECTS, WS, jährlich)..... 53

Stat-B-01: Methoden der Statistik I (6 ECTS, WS, SS)..... 145

Stat-B-02: Methoden der Statistik II (6 ECTS, WS, SS)..... 147

WiMa-B-002: Wirtschaftsmathematik: Analysis (6 ECTS, WS, SS)..... 156

KTR-MfI-2: Mathematik für Informatik 2 (Lineare Algebra) (6 ECTS, SS, jährlich).....84

## 2) A2 Fachstudium Allgemeine Informatik Grundlagen (Modulgruppe) ECTS: 42

GdI-GTI-B: Grundlagen der Theoretischen Informatik (6 ECTS, SS, jährlich)..... 49

DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software (6 ECTS, WS, jährlich)..... 32

AI-AuD-B: Algorithmen und Datenstrukturen (6 ECTS, SS, jährlich)..... 11

SWT-FPS-B: Foundations of Program Semantics (6 ECTS, WS, jährlich)..... 127

MOBI-DBS-B: Datenbanksysteme (6 ECTS, WS, SS).....109

PSI-EiRBS-B: Einführung in Rechner- und Betriebssysteme (6 ECTS, SS, jährlich)..... 117

PSI-IntroSP-B: Introduction to Security and Privacy (6 ECTS, WS, jährlich)..... 119

## 3) A3 Fachstudium Programmierung und Softwaretechnik (Modulgruppe) ECTS: 24 - 36

### a) A3 (Pflichtbereich) ECTS: 18

Bis Sommersemester 2024 wurde statt GdI-MTL-B das Modul GdI-IFP-B absolviert.

GdI-MTL-B: Modal and Temporal Logic (6 ECTS, WS, jährlich).....51

SWT-FSE-B: Foundations of Software Engineering (6 ECTS, SS, jährlich).....130

DSG-JaP-B: Java Programmierung (3 ECTS, WS, jährlich)..... 38

DSG-AJP-B: Fortgeschrittene Java Programmierung (3 ECTS, SS, jährlich)..... 30

### b) A3 (Wahlpflichtbereich) ECTS: 6 - 18

AlgoK-AK-B: Algorithmen und Komplexität (6 ECTS, SS, jährlich)..... 20

AlgoK-TAG: Tree decompositions, algorithms and games (6 ECTS, WS, jährlich).....24

DT-CPP-B: Einführung in die Systemprogrammierung in C++ (6 ECTS, WS, jährlich).....46

HCI-IS-B: Interaktive Systeme (6 ECTS, WS, jährlich)..... 59

KogSys-KI-B: Einführung in die Künstliche Intelligenz (6 ECTS, SS, jährlich)..... 93

KogSys-ML-B: Einführung in Maschinelles Lernen (6 ECTS, WS, jährlich)..... 96

VIS-GIV-B: Grundlagen der Informationsvisualisierung (6 ECTS, SS, jährlich)..... 151

xAI-MML-B: Mathematics for Machine Learning (6 ECTS, SS, jährlich)..... 158

## 4) A4 Fachstudium Komplexe und Verteilte Systeme (Modulgruppe) ECTS: 15 - 27

### a) A4 (Pflichtbereich) ECTS: 15

DSG-IDistrSys-B: Introduction to Distributed Systems (6 ECTS, SS, jährlich).....35

DSG-PKS-B: Programmierung komplexer interagierender Systeme (3 ECTS, WS, jährlich).....40

KTR-Datkomm-B: Datenkommunikation (6 ECTS, WS, jährlich).....80

### b) A4 (Wahlpflichtbereich) ECTS: 0 - 12

Das Modul GdI-MTL-B ist seit Wintersemester 2024/25 hier nicht mehr wählbar.

HCI-US-B: Ubiquitäre Systeme (6 ECTS, WS, jährlich).....64

MI-WebT-B: Web-Technologien (6 ECTS, SS, jährlich).....105

SYSNAP-SNAP-B: Systemnahe Programmierung (6 ECTS, WS, jährlich).....142

## 5) A5 Anwendungskontext Software Systems Science (Modulgruppe) ECTS: 8 - 16

Studierende, die das Modul Wirtschaftsmathematik mit 4 ECTS absolviert haben (WiMa-B-01b), müssen in dieser Modulgruppe **10 - 18 ECTS** erreichen.

Wählbar sind Module der Modulgruppe A3 gemäß der Studien- und Fachprüfungsordnung für den Bachelorstudiengang Angewandte Informatik und Module der Modulgruppe A1 gemäß der Studien- und Fachprüfungsordnung für den Bachelorstudiengang Wirtschaftsinformatik, sofern diese nicht in den Modulgruppen A3 und A4 wählbar sind. Ferner sind Module aus dem Nebenfachangebot gemäß Anhang APO GuK/Huwi 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 Überfachliche Qualifikationen (Modulgruppe) ECTS: 6 - 18

### a) A6 (Pflichtbereich) ECTS: 6

MI-WAIAI-B: Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik (3 ECTS, SS, jährlich)..... 103

PSI-EDS-B: Ethics for the Digital Society (3 ECTS, WS, jährlich)..... 115

### b) A6 (Wahlpflichtbereich) ECTS: 0 - 12

#### aa) Allgemeine Schlüsselqualifikationen (Fach) ECTS: 0 - 12

AIC-IITP-B: Internationales IT Projektmanagement (6 ECTS, SS, jährlich)..... 14

PSI-DatSchu-B: Datenschutz (3 ECTS, SS, jährlich)..... 114

KogSys-GAI-B: Genderaspekte in der Informatik (3 ECTS, SS, jährlich)..... 91

#### bb) Fremdsprachen (Fach) ECTS: 0 - 12

Im Bereich Fremdsprachen können Module gemäß dem Angebot des Sprachenzentrums der Universität Bamberg absolviert werden, ausgenommen die Module der Bereiche Deutsch als Fremdsprache und Wirtschaftsdeutsch.

## 7) A7 Seminare und Projekte (Modulgruppe) ECTS: 18 - 24

Es sind zusätzlich zum Pflichtbereich zwei Seminarmodule in den Themenbereichen A2-A4 mit jeweils 3 ECTS-Punkten sowie ein bis zwei Projektmodule mit je 6 ECTS-Punkten zu absolvieren. Mindestens ein Seminarmodul sowie ein Projektmodul müssen der Fächergruppe Informatik entstammen.

Für Studierende, die vor dem Wintersemester 21/22 ein Projekt mit 12 ECTS absolviert oder in Teilen absolviert haben, ersetzt dieses zwei Projektmodule.

Die Zulassung zur Modulprüfung setzt bei den Modulen dieser Modulgruppe eine regelmäßige Teilnahme an den zugehörigen Lehrveranstaltungen gem. § 9 Abs. 10 APO voraus.

### a) Pflichtbereich (Pflichtbereich) ECTS: 6

SWT-SWL-B: Software Engineering Lab (6 ECTS, WS, jährlich)..... 134

### b) Seminare (Wahlpflichtbereich) ECTS: 6

#### aa) Seminar(e) in Informatik (Fach) ECTS: 3 - 6

AlgoK-Sem-B: Bachelorseminar Algorithmen und Komplexitätstheorie (3 ECTS, jährlich nach Bedarf WS und SS)..... 23

DSG-Sem-B: Bachelorseminar zur Praktischen Informatik (3 ECTS, WS, SS)..... 44

DT-DB4MLKD-B: Moderne Datenbanksysteme für maschinelles Lernen und Wissensentdeckung (3 ECTS, jährlich nach Bedarf WS und SS)..... 47

GdI-Sem-B: Bachelorseminar Grundlagen der Informatik (3 ECTS, jährlich nach Bedarf WS und SS).....57

KTR-Sem-B: Bachelorseminar zu Kommunikationssystemen und Rechnernetzen (3 ECTS, jährlich nach Bedarf WS und SS)..... 89

MOBI-SEM-B: Bachelor-Seminar Mobile Software Systems (3 ECTS, WS, jährlich).....113

PSI-Sem-B: Seminar Security and Privacy Foundations (3 ECTS, WS, SS)..... 125

SWT-SEM-B: Seminar Software Engineering and Programming Languages (Bachelor) (3 ECTS, SS, jährlich)..... 132

SYSNAP-SEM-B: Seminar System Software (3 ECTS, WS, SS)..... 140

### bb) Seminar in Angewandter Informatik oder Wirtschaftsinformatik (Fach) ECTS: 0 - 3

AISE-SemCP-B: Bachelorseminar Computational Philosophy (3 ECTS, WS, jährlich).....16

AISE-SemSC-B: Bachelorseminar Smart City (3 ECTS, SS, jährlich)..... 18

CG-SemCGA-B: Seminar Computergrafik und Animation (3 ECTS, SS, jährlich)..... 26

CG-SemVRAR-B: Seminar Virtual Reality / Augmented Reality (3 ECTS, WS, jährlich)..... 27

DS-Sem-B: Bachelorseminar Dialogsysteme (3 ECTS, WS, SS)..... 28

HCI-Sem-B: Bachelorseminar Mensch-Computer-Interaktion (3 ECTS, SS, jährlich)..... 62

KInf-Seminar-B: Bachelorseminar Kulturinformatik (3 ECTS, SS, jährlich)..... 78

KogSys-Sem-B: Bachelorseminar Kognitive Systeme (3 ECTS, WS, jährlich)..... 99

MI-Sem-B: Bachelorseminar zur Medieninformatik (3 ECTS, SS, jährlich)..... 101

MII-SemHRI-B: Bachelorseminar Mensch-Roboter-Interaktion (3 ECTS, WS, jährlich)..... 108

UxD-Sem-B: Bachelorseminar User Experience and Design (3 ECTS, jährlich nach Bedarf WS oder SS).....149

VIS-Sem-B: Bachelorseminar Informationsvisualisierung (3 ECTS, WS, SS).....153

WI-Seminar-B: Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik (3 ECTS, WS, SS).....155

xAI-Sem-B1: Bachelorseminar Erklärbares Maschinelles Lernen (3 ECTS, WS, SS).....160

## c) Projekte (Wahlpflichtbereich) ECTS: 6 - 12

DSG-Project-B: Bachelorprojekt zur Praktischen Informatik (6 ECTS, SS, jährlich)..... 42

DT-Proj-B: Bachelor-Projekt: Data Engineering (6 ECTS, WS, SS).....48

GdI-Proj-B: Bachelorprojekt Grundlagen der Informatik (6 ECTS, WS, SS)..... 55

KTR-Proj: Projekt Kommunikationsnetze und -dienste (6 ECTS, WS, jährlich).....86

MOBI-Proj-B: Bachelor Project Mobile Software Systems (6 ECTS, WS, SS).....111

PSI-ProjectPAD: Project Practical Attacks and Defenses (6 ECTS, WS, SS).....122

SWT-SWP-B: Software Engineering Project (6 ECTS, SS, jährlich)..... 136

SYSNAP-Project-B: Projekt Systemnahe Programmierung (6 ECTS, WS, SS)..... 138

## 8) A8 Bachelorarbeit (Modulgruppe) ECTS: 12

SoSySc-Thesis-B: Bachelorarbeit in Software Systems Science (12 ECTS, WS, SS).....144

## 9) Module gemäß Abweichungen im Modulangebot A1 & A2 (Modulgruppe)

Um die Module

- GdI-GTI-B in A2
- DSG-EiAPS-B in A2
- PSI-EiRBS-B in A2
- GdI-MfI-1 in A1

abzuschließen, muss auf die Lehrveranstaltungen zu folgenden Modulen zurückgegriffen werden (Reihenfolge entspricht Auflistung):

Inf-DM-B: Diskrete Modellierung (9 ECTS, WS, jährlich)..... 67

Inf-Einf-B: Einführung in die Informatik (9 ECTS, WS, jährlich)..... 69

Inf-GRABS-B: Grundlagen der Rechnerarchitektur und Betriebssysteme (9 ECTS, SS, jährlich)..... 72

Inf-LBR-B: Logik und Berechenbarkeit (9 ECTS, SS, jährlich).....74

| **Modul AI-AuD-B Algorithmen und Datenstrukturen**<br>*Algorithms and Data Structures* | 6 ECTS / 180 h<br>42 h Präsenzzeit<br>138 h Selbststudium |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Andreas Henrich | |
| **Inhalte:**<br>Grundlegende Algorithmen (insbesondere Suchen, Sortieren, elementare Graphalgorithmen) und Datenstrukturen (insbesondere Listen, Hashtabellen, Bäume, Graphen) werden vorgestellt und analysiert. Konzepte der Korrektheit, Komplexität und der Algorithmenkonstruktion werden eingeführt. | |
| **Lernziele/Kompetenzen:**<br>Das Modul vermittelt Kompetenzen, Datenstrukturen und Algorithmen im Hinblick auf konkrete Anforderungen auswählen zu können, sie analysieren und durch Implementierung in einem Programm umsetzen zu können. Daneben sollen grundlegende Kompetenzen im Bereich der Algorithmenkonstruktion erworben werden. Durch die Übung soll auch die Fähigkeit zur Bewältigung von Programmieraufgaben erweitert sowie Teamarbeit geübt werden. | |
| **Sonstige Informationen:**<br>Ein Studium der Informatik erfordert grundsätzlich, sich Inhalte parallel zu den Lehrveranstaltungen praktisch und theoretisch zu erschließen (Programmierung, Formalisierung, Beweisführung). Eine aktive Teilnahme an den Übungen sowie die Bearbeitung der Übungsaufgaben ist deshalb essentiell für den Studienerfolg in diesem Modul. Der Arbeitsaufwand gliedert sich grob wie folgt:<br>• Vorlesung 21h (14 Wochen à 1,5 Stunden)<br>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30h<br>• semesterbegleitendes Üben und Bearbeiten von Übungsaufgaben und Teilleistungen: ca. 80h<br>• Übung/Tutorium 21h (14 Wochen à 1,5 Stunden)<br>• Klausur sowie Klausvorbereitung basierend auf dem erarbeiteten Stoff: ca. 30h | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Grundlegende Kenntnisse in Informatik und Programmierung wie sie im Modul Inf-Einf-B vermittelt werden sowie Basiskenntnisse der Mathematik werden vorausgesetzt, insbesondere mathematische Notationen und elementare Beweisführung.<br>Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 2. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **1. Algorithmen und Datenstrukturen**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Andreas Henrich<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Inhalte:** | |

Die Vorlesung betrachtet die zentralen Bereiche des Themengebietes Algorithmen und Datenstrukturen:

- Komplexitätsbegriffe (insb. Laufzeitkomplexität, Speicherplatzkomplexität, O-Notation)
- Korrektheit von Algorithmen
- Listen (einfach/doppelt verkettet, Stack, Queue)
- Hashverfahren
- Bäume (Datenstruktur, Traversierung, Binär-, AVL-, Suchbäume, Heap)
- Graphen (Datenstruktur, DFS-, BFS-, Dijkstra-Algorithmus, grundlegende graphentheoretische Konzepte)
- Sortieren
- Algorithmenkonstruktion

**Literatur:**
Als begleitende Lektüre wird ein Standardlehrbuch über Algorithmen und Datenstrukturen empfohlen, zum Beispiel:

- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest und Clifford Stein. Introduction to Algorithms, 4. Aufl., MIT Press, 2022
- Guter Saake und Kai-Uwe Sattler Algorithmen und Datenstrukturen: Eine Einführung mit JAVA, ISBN: 978-3864901362, 5. Aufl. 2013, 576 Seiten, dpunkt.lehrbuch
- Thomas Ottmann und Peter Widmayer: Algorithmen und Datenstrukturen, ISBN: 978-3827428035, 5. Aufl. 2012, 800 Seiten, Spektrum, Akademischer Verlag

**2. Algorithmen und Datenstrukturen** — **2,00 SWS**

**Lehrformen:** Übung
**Dozenten:** Mitarbeiter Medieninformatik
**Sprache:** Deutsch
**Angebotshäufigkeit:** SS, jährlich

**Inhalte:**
In der Übung werden Vorlesungsinhalte vertieft und deren praktische Anwendung geübt. Insbesondere werden folgende Aspekte betrachtet:

- Verständnis und Nutzung von Algorithmen
- Aufwandsbestimmung für Algorithmen
- Implementierung von Algorithmen und Datenstrukturen
- abstrakte Datentypen sowie Nutzung von Bibliotheken
- Anwendung von Prinzipien zur Algorithmenkonstruktion

**Literatur:**
siehe Vorlesung; weitere Literaturempfehlungen werden in der Übung bekanntgegeben

**Prüfung**
schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten
**Beschreibung:**

| | |
|---|---|
| Gegenstand der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich Übungsaufgaben; siehe unten). Die Prüfungsdauer beinhaltet eine Lesezeit von 15 Minuten, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.<br><br>In der Klausur können 90 Punkte erzielt werden. Ferner werden optionale semesterbegleitende Studienleistungen zur Notenverbesserung im Rahmen des Übungsbetriebs angeboten. Dabei können durch die Abgabe bzw. Vorstellung von Lösungen zu Übungsaufgaben Bonuspunkte erzielt werden. Ist die Klausur bestanden, werden die in den semesterbegleitenden Studienleistungen erzielten Punkte zu der in der Klausur erreichten Punktzahl hinzuaddiert. Die im Einzelnen zu erbringenden optionalen Studienleistungen, deren jeweilige Bearbeitungsdauer bzw. Bearbeitungsfrist sowie die durch Studien- und Prüfungsleistungen jeweils und insgesamt erreichbare Punktzahl werden zu Beginn des Semesters in der Übung und im Kurs im Virtuellen Campus bekanntgegeben. Die Note 1,0 ist auch ohne Punkte aus den semesterbegleitenden Studienleistungen erreichbar. | |

| **Modul AIC-IITP-B Internationales IT Projektmanagement**<br>*International IT Project Management* | 6 ECTS / 180 h |
|---|---|
| (seit SS25)<br>Modulverantwortliche/r: Prof. Dr. Milad Mirbabaie | |
| **Inhalte:**<br>Der Kurs vermittelt Fertigkeiten zum Management von und zur Mitarbeit in internationalen IT-Projekten. Behandelt werden alle Stufen entlang des Projektlebenszyklus von der initialen Betrachtung über die Planung, die Ausführung, das Projekt-Controlling und den Projektabschluss. Dabei finden sowohl traditionelle Vorgehensmodelle (z.B. Wasserfall- und V-Modelle) als auch agile Methoden (Scrum) Berücksichtigung. Behandelt werden ebenfalls Besonderheiten internationaler Teams und räumlich verteilter Projekte.<br>Der Kurs verbindet die Vermittlung von Wissen zu Werkzeugen, etablierten Lösungstechniken, Bewertungsschemata etc. mit theoretischen Grundlagen und einer kritischen Auseinandersetzung bzgl. der Stärken und Grenzen der Ansätze. | |
| **Lernziele/Kompetenzen:**<br>Studierende sollen nach erfolgreichem Abschluss der Veranstaltung in der Lage sein, IT-Projekte in kleinen und großen Organisationen zu initiieren, zu planen, zu leiten und zu überwachen. Dazu können sie aus den behandelten Methoden und Vorgehensmodellen geeignete auswählen, die Wahl begründen, die Ansätze an neue und bisher unbekannte Problemstellungen anpassen und basierend auf den theoretischen Grundlagen fundiert und adäquat weiterentwickeln. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Keine | |
| **Empfohlene Vorkenntnisse:**<br>Keine | **Besondere Bestehensvoraussetzungen:**<br>Keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Internationales IT Projektmanagement**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Milad Mirbabaie<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **4,00 SWS** |
| **Lernziele:**<br>Studierende sollen nach erfolgreichem Abschluss der Veranstaltung in der Lage sein, IT-Projekte in kleinen und großen Organisationen zu initiieren, zu planen, zu leiten und zu überwachen. Dazu können sie aus den behandelten Methoden und Vorgehensmodellen geeignete auswählen, die Wahl begründen, die Ansätze an neue und bisher unbekannte Problemstellungen anpassen und basierend auf den theoretischen Grundlagen fundiert und adäquat weiterentwickeln. | |
| **Inhalte:** | |

| | |
|---|---|
| Der Kurs vermittelt Fertigkeiten zum Management von und zur Mitarbeit in internationalen IT-Projekten.<br>Behandelt werden alle Stufen entlang des Projektlebenszyklus von der initialen Betrachtung über die Planung, die Ausführung, das Projekt-Controlling und den Projektabschluss. Dabei finden sowohl traditionelle Vorgehensmodelle (z.B. Wasserfall- und V-Modelle) als auch agile Methoden (Scrum) Berücksichtigung.<br>Behandelt werden ebenfalls Besonderheiten internationaler Teams und räumlich verteilter Projekte.<br>Der Kurs verbindet die Vermittlung von Wissen zu Werkzeugen, etablierten Lösungstechniken,<br>Bewertungsschemata etc. mit theoretischen Grundlagen und einer kritischen Auseinandersetzung bzgl. der Stärken und Grenzen der Ansätze.<br>**Literatur:**<br>Wird im Kurs bekannt gegeben. | |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten | |

| **Modul AISE-SemCP-B Bachelorseminar Computational Philosophy**<br>*Bachelorseminar Computational Philosophy* | 3 ECTS / 90 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Christoph Benzmüller<br>Weitere Verantwortliche: Prof. Dr. Andrea Vestrucci | |
| **Inhalte:**<br>Gottfried Wilhelm Leibniz had a vision: human thoughts made computable. What is real, what is good, what is beautiful, what is personal identity… all facets of human thinking treated and organised as elements of calculation. This vision had a tremendous multi-part impact which the seminar shall follow: It fostered the development of machines to calculate (“ordinateur”, the French word for “computer,” is what makes order out of chaos); It influenced the so-called analytical approach in philosophy, treating propositions formally; And it established the basis for the science “prior to all others” (K. Gödel), i.e., mathematical logic. This seminar in Computational Philosophy will embrace these three aspects. It will study how, and how far, thinking and algorithm are one thing, and can impact each other. It will ask why Leibniz’s vision has not become reality: What are the computational limits of human thought? What are the (philosophical) limits of computation? And, more importantly, what is left to do? What are the future directions on the path of making our thinking computable, and a machine thinking? | |
| **Lernziele/Kompetenzen:**<br>Via the application of AI programs, we will explore the above questions and try to formulate our answers. We will, for instance, discuss current interpretations of AI limits; we will deepen the relationship between Gödel’s incompleteness theorems and Turing’s halting problem; we will explore metaphysical arguments, belief changes, and ethical problems in an automated reasoning environment. | |
| **Sonstige Informationen:**<br>The main language of instruction in this course is English. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Basic knowledge on AI, theoretical computer science and cognitive science; some background in theoretical philosophy is also useful. | |
| **Empfohlene Vorkenntnisse:**<br>keine | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Computational Philosophy**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Christoph Benzmüller<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |
| **Lernziele:**<br>Via the application of AI programs, we will explore the above questions and try to formulate our answers. We will, for instance, discuss current interpretations of AI limits; we will deepen the relationship between Gödel’s incompleteness theorems | |

| | |
|---|---|
| and Turing's halting problem; we will explore metaphysical arguments, belief changes, and ethical problems in an automated reasoning environment.<br><br>**Inhalte:**<br>Gottfried Wilhelm Leibniz had a vision: human thoughts made computable. What is real, what is good, what is beautiful, what is personal identity… all facets of human thinking treated and organised as elements of calculation. This vision had a tremendous multi-part impact which the seminar shall follow: It fostered the development of machines to calculate ("ordinateur", the French word for "computer," is what makes order out of chaos); It influenced the so-called analytical approach in philosophy, treating propositions formally; And it established the basis for the science "prior to all others" (K. Gödel), i.e., mathematical logic. This seminar in Computational Philosophy will embrace these three aspects. It will study how, and how far, thinking and algorithm are one thing, and can impact each other. It will ask why Leibniz's vision has not become reality: What are the computational limits of human thought? What are the (philosophical) limits of computation? And, more importantly, what is left to do? What are the future directions on the path of making our thinking computable, and a machine thinking?<br><br>**Literatur:**<br>to be announced in lecture course | |
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Continuous attendance in the seminar sessions is mandatory, cf. §9 (10) APO.<br>**Beschreibung:**<br>The module examination consists of two parts, a term paper (in English) and a talk (in English). | |

| **Modul AISE-SemSC-B Bachelorseminar Smart City**<br>*Bachelorseminar Smart City* | | 3 ECTS / 90 h |
|---|---|---|
| (seit SS23)<br>Modulverantwortliche/r: Prof. Dr. Christoph Benzmüller | | |
| **Inhalte:**<br>The course, titled "AI Ethics in the Smart City," focuses on the practical aspects of AI ethics within the context of smart cities. The course includes introductory sessions on ethical issues in AI applications specific to smart city environments, such as projects of AI system integration with the Klinikum Bamberg, and discussions and implementation sessions at UniBamberg. | | |
| **Lernziele/Kompetenzen:**<br>Equip students with a comprehensive understanding of real-world challenges in AI ethics, particularly in the context of a city's digital transformation. Provide tools and resources to enable students to contribute to cross-disciplinary initiatives and potentially engage further with projects or Master's theses. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Students would benefit from being open to and skilled in interdisciplinary work, combining technology with real-world applications. Ability to engage in critical thinking and problem-solving, especially in applying theoretical knowledge to practical situations, would be highly beneficial. Effective communication skills, particularly in English (and possibly German). Practical skills in relevant software or programming languages might be beneficial, but not mandatory. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **AI Ethics in the Smart City**<br>**Lehrformen:** Blockseminar<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Inhalte:**<br>The Blockseminar offers an immersive experience for students to delve into the practical aspects of AI ethics within the context of Smart Cities, by focusing on an existing collaborative project with Klinikum Bamberg. This project focuses on developing an AI Ethical Decider tailored for Spontaneous Breathing Trials in Intensive Care Unit patients, who are on mechanical ventilation.<br>The primary objectives are threefold:<br>1. To equip students with a comprehensive understanding of the real-world challenges in AI ethics, especially in the context of a city s digital transformation;<br>2. To expose students to the synergies between healthcare systems and AI research, fostering an interdisciplinary approach; | |

| | |
|---|---|
| 3. To provide tools and resources enabling students to contribute to this cross-disciplinary initiative.<br>Additionally, the Blockseminar aims to ignite student interest in further engagement with this project, e.g. in the form of Bachelor or Master theses. | |
| **Prüfung**<br>Hausarbeit mit Referat | |

| **Modul AlgoK-AK-B Algorithmen und Komplexität**<br>*algorithms and complexity* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Isolde Adler | | |
| **Inhalte:**<br>Algorithms and problem solving lie at the heart of computer science.<br>Given an algorithmic problem, such as the Traveling Salesperson Problem, how can we design an efficient algorithm? Once we found an algorithm that solves the problem correctly, can we be sure that the resources, such as running time, storage space (and related: energy), required by this algorithm are really necessary for solving the problem? Perhaps we can do better? | | |
| **Lernziele/Kompetenzen:**<br>Demonstrate an understanding of what constitutes an efficient and an inefficient solution to a computational problem,<br>- Analyse the efficiency of algorithms,<br>- Evaluate and justify appropriate ways to provide efficient solutions for computational problems,<br>- Identify and apply different design principles in the design of algorithms,<br>- Describe efficient algorithms for a range of computational problems, along with their computational complexity,<br>- Describe the use of complexity classes and the relations between them,<br>- Articulate the key concepts and critically evaluate approaches in a clear and rigorous manner,<br>- Appreciate and understand in-depth the role of proofs in the area of algorithm design,<br>- Recognise how the methods learned can be extended and used to solve other problems. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Keine | | |
| **Empfohlene Vorkenntnisse:**<br>Algorithms and data structures, basic knowledge of computability theory, proof techniques. Good English language skills. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** |

| | | |
|---|---|---|
| | | 1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **AlgoK-AK-B (Algorithmen und Komplexität)**<br>**Lehrformen:** Vorlesung und Übung<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Lernziele:**<br>Demonstrate an understanding of what constitutes an efficient and an inefficient solution to a computational problem,<br><br>- Analyse the efficiency of algorithms,<br><br>- Evaluate and justify appropriate ways to provide efficient solutions for computational problems,<br><br>- Identify and apply different design principles in the design of algorithms,<br><br>- Describe efficient algorithms for a range of computational problems, along with their computational complexity,<br><br>- Describe the use of complexity classes and the relations between them,<br><br>- Articulate the key concepts and critically evaluate approaches in a clear and rigorous manner,<br><br>- Appreciate and understand in-depth the role of proofs in the area of algorithm design,<br><br>- Recognise how the methods learned can be extended and used to solve other problems. | |
| **Inhalte:**<br>Algorithms and problem solving lie at the heart of computer science.<br>Given an algorithmic problem, such as the Traveling Salesperson Problem, how can we design an efficient algorithm? Once we found an algorithm that solves the problem correctly, can we be sure that the resources, such as running time, storage space (and related: energy), required by this algorithm are really necessary for solving the problem? Perhaps we can do better? | |

| | |
|---|---|
| **Prüfung**<br>mündliche Prüfung<br>**Beschreibung:**<br>Die Prüfungsdauer wird in der ersten LV bekannt gegeben | |

| **Modul AlgoK-Sem-B Bachelorseminar Algorithmen und Komplexitätstheorie**<br>*Bachelor Seminar Algorithms and Complexity Theory* | 3 ECTS / 90 h |
|---|---|
| (seit WS22/23)<br>Modulverantwortliche/r: Prof. Dr. Isolde Adler | |
| **Inhalte:**<br>Im Seminarmodul werden wechselnde Themen im Bereich der Algorithmen und Komplexitätstheorie angeboten. | |
| **Lernziele/Kompetenzen:**<br>Fähigkeit zur selbständigen Erarbeitung von Inhalten aus der aktuellen wissenschaftlichen Literatur; Fähigkeit, komplexe Problemlösungsansätze schriftlich und mündlich zu vermitteln. Förderung der wissenschaftliche Neugier und die Ausbildung einer selbstbewussten und forschenden Einstellung zur Technik. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Grundlagen der Theoretischen Informatik, Mathematik für Informatik 1, Algorithmen und Datenstrukturen, gute Englischkenntnisse | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** jährlich nach Bedarf WS und SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Algorithmen und Komplexitätstheoerie**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Isolde Adler<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** jährlich nach Bedarf WS und SS<br>**Inhalte:**<br>Das AlgoK-Seminar wird zu semesterweise wechselnden Themen angeboten.<br>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.<br>**Literatur:**<br>Literatur wird bei Ankündigung bzw. zu Beginn des Seminars bekanntgegeben. | **2,00 SWS** |

| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an den Terminen der Lehrveranstaltungen.<br>**Beschreibung:**<br>Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben. | |
|---|---|

| **Modul AlgoK-TAG Tree decompositions, algorithms and games**<br>*Baumzerlegungen, Algorithmen und Spiele* | 6 ECTS / 180 h |
|---|---|
| (seit WS23/24)<br>Modulverantwortliche/r: Prof. Dr. Isolde Adler | |

**Inhalte:**
Many classical algorithmic problems on graphs are hard, e.g. NP-hard, in general. However, they lie at the core of many applications, so they need to be solved in practice. These problems include the famous Graph Colouring Problem, and problems such as Hamiltonian Cycle, Independent Set, Dominating Set, Vertex Cover and many more.
Ideally, we would like to solve these problems exactly and efficiently.
Indeed, many problems become solvable in linear time if we only allow trees as inputs. This observation is the starting point of the module. We then identify more general classes of input graphs that allow solving many problems efficiently.
For this we make use of so-called tree decompositions of graphs. Tree decompositions allow us to obtain "tree like" graphs that are more general than trees but maintain the favourable algorithmic properties of trees.
In the first part of the module we study tree decompositions via a cops-and-robber game played on graphs, where the winning strategies for the cops yield the desired decompositions. We then develop algorithms for tree decompositions and algorithms to solve problems efficiently making use of tree decompositions.
In the second part of the module we introduce monadic second order logic (MSO) on graphs and we prove a famous theorem by Bruno Courcelle that shows how to solve **all** problems expressible in MSO efficiently on "tree-like" graphs. This includes all aforementioned algorithmic problems.
We make links to state-of-the-art research in the area and to practical applications, e.g. in compiler construction.

**Lernziele/Kompetenzen:**
On completion of this module, students should
- be familiar with classical NP-complete problems on graphs and how to solve them efficiently on trees using dynamic programming
- be able to demonstrate an in-depth understanding of tree decompositions, algorithms for computing tree decompositions, and algorithms on tree decompositions
- be able to demonstrate an in-depth understanding of the cops-and-robber game, its game theoretic properties and its connection to tree decompositions
- be able to design algorithms for the relevant problems, including analysis of runtime and correctness proofs
- be able to explain the main results covered by the module, in particular Courcelle's Theorem, demonstrating an understanding by discussing examples and knowing the main proof ideas
- be aware of the practical applications and limitations of the results
- appreciate and understand in-depth the role of proofs in the area of algorithm design

- recognise how the methods learned can be extended and used to solve other problems.

**Sonstige Informationen:**
The workload for this module is approxmately structured as follows:

- Participation in lectures and tutorials: 45 hrs
- Preparing and revising the lectures and tutorials: 60 hours
- Solving the worksheets: 45 hrs
- Exam preparation: 30 hrs

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:** | **Besondere Bestehensvoraussetzungen:** |
|---|---|
| Prerequisites: Algorithms and data structures, basic knowledge of predicate logic, proof techniques, interest in combinatorial games on graphs.<br>Good English language skills. | keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** ab dem 3. | **Minimale Dauer des Moduls:** 1 Semester |
|---|---|---|

**Lehrveranstaltungen**

**Tree Decompositions, Algorithms and Games** — **4,00 SWS**

**Lehrformen:** Vorlesung und Übung
**Sprache:** Englisch/Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**
The lectures introduce the topics, providing an in-depth explanation including motivation, intuition, examples and proofs, as well as tools, techniques and applications.

The tutorials consist of hands-on problem solving, including exam-style problems.

**Literatur:**
- Reinhard Diestel: Graph Theory, Springer 2017
- Jörg Flum, Martin Grohe: Parameterized Complexity Theory, Springer 2010
- Anthony Bonato, Richard J. Nowakowski: The Game of Cops and Robbers on Graphs, American Mathematical Society, 2011
- Bruno Courcelle, Joost Engelfriet: Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach (Encyclopedia of Mathematics and its Applications Book 138), Cambridge University Press, 2012

**Prüfung**
Sonstiges / Prüfungsdauer: 90 Minuten
**Beschreibung:**
Oral exam (30 minutes) or written exam (90 minutes).

Depending on the number of participants, the exam will either be an oral exam or a written exam. The mode of examination will be communicated in the first lecture.

| **Modul CG-SemCGA-B Seminar Computergrafik und Animation**<br>*Seminar Computer Graphics and Animation* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Sophie Jörg | | |
| **Inhalte:**<br>Computer generated content is very common, for example, in movies, virtual worlds or educational applications. In this seminar we will be learning about the foundations and current approaches of computer graphics research. Your task will include reading and discussing foundational and contemporary papers in the field. You will then create a presentation and a report focused on a particular aspect of the field. | | |
| **Lernziele/Kompetenzen:**<br>Participants will practice methods for reading, discussing, and presenting scintific literature about computer graphics systems, algorithms and technologies as well as methods for writing an academic report. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Modul CG-CGA-B Computergrafik und Animation | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Seminar Grundlagen der Computergrafik und Animation**<br>**Lehrformen:** Seminar<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |

| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 2 Monate | |
|---|---|

| | |
|---|---|
| **Modul CG-SemVRAR-B Seminar Virtual Reality / Augmented Reality**<br>*Seminar Virtual Reality / Augmented Reality* | 3 ECTS / 90 h |
| (seit WS23/24)<br>Modulverantwortliche/r: Prof. Dr. Sophie Jörg | |
| **Inhalte:**<br>Virtual reality (VR) and augmented reality (AR) are becoming increasingly popular. In such worlds we are represented by virtual characters called avatars. In this seminar, we will explore topics related to virtual reality, augmented reality, and avatars. We will read and critically discuss state-of-the-art research, learn about experiment design in the field, identify open questions, and think about how to design experiments to answer these.<br>Participants will choose selected subtopics and present them as well as create a written seminar report. | |
| **Lernziele/Kompetenzen:**<br>Participants will practice methods for academic writing and presentations as well as get an understanding of current research topics in virtual reality and augmented reality. | |
| **Sonstige Informationen:**<br>Typical work load:<br>• Meetings and talks: ~20h<br>• Literature search and reading: ~25h<br>• Preparation of presentation: ~15h<br>• Written report: ~30h | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in virtual reality or user studies is advantageous. | **Besondere Bestehensvoraussetzungen:**<br>keine |

| | | |
|---|---|---|
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Seminar Virtual Reality / Augmented Reality**<br>**Lehrformen:** Seminar<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 2 Monate<br>**Beschreibung:**<br>Participants will give a talk on a selected topic and submit a written seminar report. | |

| **Modul DS-Sem-B Bachelorseminar Dialogsysteme**<br>*Bachelor Seminar Dialogue Systems* | | 3 ECTS / 90 h |
|---|---|---|
| (seit SS23)<br>Modulverantwortliche/r: Prof. Dr. Stefan Ultes | | |
| **Inhalte:**<br>Digitale persönliche Assistenten wie Siri, Google Assistant und Alexa sind aus der heutigen Zeit nicht mehr wegzudenken, was sie nicht zuletzt ihrer Fähigkeit der sprachbasierten Interaktion verdanken. Diese Systeme werden dem Bereich Conversational AI zugeordnet, die dahinterstehende Technologie Dialogsysteme genannt. Im diesem Modul werden aktuelle und grundlegenderen Forschungsarbeiten aus dem Bereich Dialogsysteme und Conversational AI behandelt, wissenschaftlich aufbereitet und präsentiert. | | |
| **Lernziele/Kompetenzen:**<br>Folgende Kompetenzen sollen durch diese Veranstaltung vermittelt bzw. vertieft werden. Die Teilnehmenden<br>• lernen sich selbständig Fachinhalte zu recherchieren und sich einzuarbeiten,<br>• üben und verstehen Methoden professioneller schriftlicher und mündlicher Kommunikation,<br>• lernen, über Systeme und Methoden im Bereich Dialogsysteme zu diskutieren und diese zu evaluieren,<br>• entwickeln ein tiefergehendes Verständnis der einzelnen Themen, deren möglichen Einsatz und deren Grenzen. | | |
| **Sonstige Informationen:**<br>Die Themen werden allein oder zu zweit bearbeiten.<br>Die Arbeitsumfänge gestalten sich typischerweise wie folgt:<br>• Gruppentreffen und Diskussionen: ~20h<br>• Literaturrecherche: ~25h<br>• Vorbereitung der Präsentation: ~15h<br>• Schriftliche Ausarbeitung: ~30h | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Für einige der Themen sind solide Kenntnisse im Bereich maschinelles Lernen/KI zum Verständnis notwendig/hilfreich; Einführung in die KI [AI-KI-B] | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Dialogsysteme**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Stefan Ultes<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS | **2,00 SWS** |
| **Lernziele:** | |

| | |
|---|---|
| siehe Modulbeschreibung<br><br>**Inhalte:**<br>Im Seminar werden verschiedene Themen rund um Dialogsysteme und Conversational AI behandelt, die jedes Semester neu ausgewählt werden. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 15 Minuten<br>**Beschreibung:**<br>Die Inhalte des Referats und der Hausarbeit befassen sich mit einem zugewiesenen Thema und relevanter verwandter Themen. Die Bearbeitungsdauer der Hausarbeit wird zu Beginn des Seminars bekannt gegeben. | |

| **Modul DSG-AJP-B Fortgeschrittene Java Programmierung**<br>*Advanced Java Programming* | 3 ECTS / 90 h<br>23 h Präsenzzeit<br>67 h Selbststudium |
|---|---|
| (seit WS16/17)<br>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:

- 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.

**Sonstige Informationen:**

Der Arbeitsaufwand von 90 Std. gliedert sich in

- 22.5 Std. Teilnahme an der Praktischen Übung
- 55 Std. Bearbeiten der Programmieraufgabe (Assignment)
- 12 Std. Vorbereitung auf das Abschlusskolloquium
- 0.5 Std. Abschlusskolloquium inklusive Warten auf Ergebnis usw.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

keine

| **Empfohlene Vorkenntnisse:** | **Besondere Bestehensvoraussetzungen:** |
|---|---|
| Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Algorithmik und Softwareentwicklung, wie sie z.B. im Modul DSG-EiAPS-B vermittelt werden.<br><br>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen | keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>2. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| Lehrveranstaltungen | |
|---|---|
| **Fortgeschrittene Java Programmierung**<br>**Lehrformen:** kein Typ gewählt, Übung<br>**Dozenten:** Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>vgl. Modulbeschreibung<br>**Inhalte:**<br>vgl. Modulbeschreibung<br>**Literatur:**<br>Jedes weiterführende Buch zu Java ist verwendbar. | **2,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Beschreibung:**<br>Die zu Beginn des Semesters ausgegebene Programmieraufgabe (Assignment) wird 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 der Programmieraufgabe vor, erläutern diese und beantworten Fragen zu ihrer Lösung und den dabei verwendeten Techniken und Sprachkonstrukten. | |

| **Modul DSG-EiAPS-B Einführung in Algorithmen, Programmierung und Software**<br>*Introduction to Algorithms, Programming and Software* | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|
| (seit WS20/21)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | |

**Inhalte:**

Das Modul gibt einen ersten Einblick in die Informatik aus Sicht der Entwicklung von Algorithmen und deren Realisierung durch Programme in imperativen Programmiersprachen (am Beispiel der Sprache ANSI-C) sowie einen Ausblick auf die Problematik der Softwareentwicklung. Behandelt werden die Grundprinzipien der Informatik zu:

- 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,
- Techniken zur Spezifikation, zur Datenabstraktion und funktionalen Abstraktion, z.B. Abstrakte Datentypen, sowie
- einfache Beschreibungsmechanismen für Sprachen wie Grammatiken (Typ 2/3) und Endliche Automaten

Alle wichtigen Begriffe werden am Beispiel der Programmiersprache ANSI-C veranschaulicht, so dass damit auch die Grundlagen imperativer Programmiersprachen eingeführt werden. Dazu gehören insbesondere die für alle Programmiersprachen wesentlichen Konzepte wie

- Wertebereiche, Namensräume und deren Realisierung durch Speichermodelle,
- Syntax vs. statische Semantik vs. dynamische Semantik,
- Seiteneffekte durch Zuweisungen vs. Berechnung von Funktionen (Parameter, Resultate),
- Call-by-Value vs. Call-by-Reference,
- Kontroll- und Datenfluss in einem Programm, bei Funktionsaufrufen usw.,
- Iteration vs. Rekursion, sowie
- Konzepte zur Strukturierung von Programmen.

Neben der Diskussion der verschiedenen Konzepte werden auch die wichtigsten Aspekte durch praktisches Programmieren eingeübt.

**Lernziele/Kompetenzen:**

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, aus einem Basisrepertoire an Möglichkeiten jeweils 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 grundsätzliche Arbeitsweise von Programmiersprachen und können die wesentlichen Schritte der Softwareentwicklung nachvollziehen. Studierende können einfache Problemstellungen beschreiben, algorithmische Lösungen dazu entwickeln und diese auch mittels einfacher Datenstrukturen in konkreten imperativen Programmiersprachen umsetzen.

**Sonstige Informationen:**
Auch wenn das Modul organisatorisch unabhängig von der Einführung in Java durch das ebenfalls jeweils im Wintersemester angebotene Modul DSG-JaP-B ist und beide Module auch bei entsprechenden Vorkenntnissen unabhängig voneinander absolviert werden können, **wird Studienanfängerinnen und -anfängern dringend empfohlen, beide Module im gleichen Semester zu bearbeiten, d.h. bei Studienbeginn zum Wintersemester im 1. Fachsemester und bei Studienbeginn zum Sommersemester jeweils im 2. Fachsemester.**

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:** | **Besondere Bestehensvoraussetzungen:** |
|---|---|
| Die Veranstaltung hat als grundlegende Einführungsveranstaltung in das Gebiet der Informatik weder Inhalte anderer Lehrveranstaltungen noch Informatikkenntnisse oder Programmierkenntnisse zur Voraussetzung. *Insbesondere ist das Modul PSI-EiRBS-B, das regelmäßig im Sommersemester angeboten wird,* ***keine*** *Voraussetzung für DSG-EiAPS-B.* | keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** ab dem 1. | **Minimale Dauer des Moduls:** 1 Semester |
|---|---|---|

**Lehrveranstaltungen**

**1. Einführung in Algorithmen, Programmierung und Software** — **2,00 SWS**

**Lehrformen:** Vorlesung
**Dozenten:** Prof. Dr. Guido Wirtz
**Sprache:** Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Lernziele:**
vgl. Modulbeschreibung

**Inhalte:**
vgl. Modulbeschreibung

**Literatur:**
Jede Einführung in die Informatik 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, die zum Teil andere Schwerpunkte setzen, nicht unbedingt die gleichen Themen behandeln, bei gleichen Themen andere Herangehensweisen an das jeweilige Thema wählen und natürlich zum Teil andere Schreibweisen usw. benutzen:

- Heinz-Peter Gumm, Manfred Sommer: Einführung in die Informatik. Oldenbourg Verlag, 2013 (10th)
- Herbert Klaeren, Michael Sperber: Die Macht der Abstraktion - Einführung in die Programmierung. Teubner, 2007 (1th)
- Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, Shriram Krishnamurthi: How to Design Programs - An Introduction to Computing and Programming. The MIT Press 2001 (online verfügbar)

| | |
|---|---|
| • Helmut Herold, Bruno Lurz, Jürgen Wohlrab: Grundlagen der Informatik. Pearson Higher Education, 2012 (2nd)<br>• Barbara Liskov with John Guttag: Program Development in Java. Addison-Wesley, 2001 | |
| **2. Einführung in Algorithmen, Programmierung und Software**<br>**Lehrformen:** Übung<br>**Dozenten:** Mitarbeiter Praktische Informatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Lernziele:**<br>vgl. Vorlesung<br>**Inhalte:**<br>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.<br>**Literatur:**<br>vgl. Vorlesung | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>Klausur zum Stoff des gesamten Moduls, also der Vorlesung und Übung zur DSG-EiAPS-B. Bestehen der Klausur durch Erreichen von 50% der maximal erreichbaren Punkte. | |

| **Modul DSG-IDistrSys-B Introduction to Distributed Systems**<br>*Introduction to Distributed Systems* | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | |
| **Inhalte:**<br>Nowadays infrastructure and business relies more or less on distributed systems of various flavors. Most of our civilization would not work any more if all distributed systems would fail. So, that should be a good reason for anyone planning to work in the context of IT to learn at least about the characteristics and basic issues of such systems. 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. Students are required to work (in groups) on an assignment using different technologies in order to combine the theoretical concepts with practical experience and ... Yes, we program! | |
| **Lernziele/Kompetenzen:**<br>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. | |
| **Sonstige Informationen:**<br>The language of instruction in this course is English.<br>The overall workload of 180h for this module consists of:<br>• weekly classes: 22.5h<br>• tutorials: 22.5h<br>• Work on assignment: 75h<br>• Literature study 20h<br>• preparation for and time of the final exam: 40h<br>This course is intended for 2nd/3rd year bachelor students. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>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 PSI-IntroSP-B and SYSNAP-GRABS-B. Preferable also basic knowledge about multithreading and synchronization.<br>Modul Programmierung komplexer interagierender Systeme (DSG-PKS-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** |

| | | |
|---|---|---|
| | ab dem 4. | 1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **1. Introduction to Distributed Systems**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Guido Wirtz<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>c.f. overall module description<br>**Inhalte:**<br>c.f. overall module description<br>**Literatur:**<br>• George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems - Concepts and Design. Pearson Education UK, 2011 (5. edition); ISBN: 9780273760597<br>• Andrew Tanenbaum, Marten van Steen: Distributed Systems - Principles and Paradigms, 2017 (3rd edition) | **2,00 SWS** |
| **2. Tutorial Introduction to Distributed Systems**<br>**Lehrformen:** Übung<br>**Dozenten:** Mitarbeiter Praktische Informatik<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>c.f. overall module description<br>**Inhalte:**<br>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 assignment. | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Beschreibung:**<br>Oral examination concerning the topics discussed in the lecture, exercises and assignment. **Students may choose English or German as the language for the written assignment and oral examination.** Examinations will take place at the end of the summer term or at the begin of the winter term (students may choose one of them).<br>Students are assumed to work on a programming assignment ('schriftliche Hausarbeit') during the semester that is introduced at the beginning of the semester and uses the most important technologies discussed during the semester. | |

| | |
|---|---|
| **Note:** Without working on the programming assignment over the term students may run into problems during their oral examination (Kolloquium) as we discuss questions concerning topics from the lectures as well as from the assignment; questions about the assignment are based on the assignment solution programmed by the students. | |

| **Modul DSG-JaP-B Java Programmierung**<br>*Programming in Java* | 3 ECTS / 90 h<br>30 h Präsenzzeit<br>60 h Selbststudium |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | |
| **Inhalte:**<br>Die Programmiersprache Java wird als imperative und auch objektorientierte Sprache in ihren wichtigsten Konzepten und Bestandteilen eingeführt. Der Schwerpunkt liegt dabei auf dem Kern der Sprache und der praktischen Anwendung der Sprache zur Erstellung einfacher Programme, dem sog. Programmieren im Kleinen. Dazu werden die relevanten Konzepte zur Erstellung einfacher Programme (Schnittstellen, Klassen und Testklassen) zur Manipulation von grundlegenden Datentypen und einfachen Datenstrukturen (Felder, einfache Listen) eingeführt und anhand von Beispielen und durch einfache Programmieraufgaben erläutert. Zudem wird die Verwendung der in Java vorhandenen Techniken zur Umsetzung objektorientierten Programmierens wie z.B. Typisierung und Vererbung sowie Mechanismen zur Abstraktion und Strukturierung wie z.B. Schnittstellen und Paket-Strukturen und einfache Fehlerbehandlung (Exceptions) diskutiert. | |
| **Lernziele/Kompetenzen:**<br>Studierende sollen mit der Programmiersprache Java und einer einfachen Programmierumgebung (Editor, Compiler, Dokumentations-Werkzeug) soweit praktisch umgehen können, dass sie aus der Beschreibung einfacher Probleme selbständig ein Lösungsverfahren entwickeln und durch Wahl geeigneter Daten- und Kontrollstrukturen in ein funktionsfähiges Java-Programm umsetzen, übersetzen und nach Auswahl geeigneter Testdaten testen können. Zusätzlich sollen die von Java angebotenen - für objektorientierte Sprachen typische - Strukturierungs- und Abstraktionstechniken wie Sichtbarkeitsbereiche, Schnittstellen und implementierende Klassen, einfache Vererbung, Programme bestehend aus mehreren Klassen sowie Verwendung eigener und vorgegebener Paketstrukturen sinnvoll eingesetzt werden können. | |
| **Sonstige Informationen:**<br>Auch wenn das Modul organisatorisch unabhängig von der grundlegenden Einführung in die Informatik durch das Modul PSI-IntroSP-B ist und bei entsprechenden Vorkenntnissen auch ohne die PSI-IntroSP-B absolviert werden kann, wird Studienanfängerinnen und -anfängern **dringend empfohlen, beide Module im gleichen Semester zu bearbeiten**, d.h. bei Studienbeginn zum Wintersemester im 1. Fachsemester und bei Studienbeginn zum Sommersemester jeweils im 2. Fachsemester. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Grundlagende Kenntnisse in einfachen Algorithmen und Kontrollstrukturen sowie den Grundprinzipien der Informatik, wie sie z.B. in den ersten Sitzungen der Vorlesung zum Modul PSI-IntroSP-B vermittelt werden, werden vorausgesetzt. | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 1. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Java Programmierung**<br>**Lehrformen:** Vorlesung, Übung | **2,00 SWS** |

**Dozenten:** Prof. Dr. Guido Wirtz
**Sprache:** Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Lernziele:**
Studierende sollen mit der Programmiersprache Java und einer einfachen Programmierumgebung (Editor, Compiler, Dokumentations-Werkzeug) soweit praktisch umgehen können, dass sie aus der Beschreibung einfacher Probleme selbständig ein Lösungsverfahren entwickeln und durch Wahl geeigneter Daten- und Kontrollstrukturen in ein funktionsfähiges Java-Programm umsetzen, übersetzen und nach Auswahl geeigneter Testdaten testen können. Zusätzlich sollen die von Java angebotenen - für objektorientierte Sprachen typische
- Strukturierungs- und Abstraktionstechniken wie Sichtbarkeitsbereiche, Schnittstellen und implementierende Klassen, einfache Vererbung, Programme bestehend aus mehreren Klassen sowie Verwendung eigener und vorgegebener Paketstrukturen sinnvoll eingesetzt werden können.

**Inhalte:**
Die Programmiersprache Java wird als imperative und auch objektorientierte Sprache in ihren wichtigsten Konzepten und Bestandteilen eingeführt. Der Schwerpunkt liegt dabei auf dem Kern der Sprache und der praktischen Anwendung der Sprache zur Erstellung einfacher Programme, dem sog. Programmieren im Kleinen. Dazu werden die relevanten Konzepte zur Erstellung einfacher Programme (Schnittstellen, Klassen und Testklassen) zur Manipulation von grundlegenden Datentypen und einfachen Datenstrukturen (Felder, einfache Listen) eingeführt und anhand von Beispielen und durch einfache Programmieraufgaben erläutert. Zudem wird die Verwendung der in Java vorhandenen Techniken zur Umsetzung objektorientierten Programmierens wie z.B. Typisierung und Vererbung sowie Mechanismen zur Abstraktion und Strukturierung wie z.B. Schnittstellen und Paket-Strukturen und einfache Fehlerbehandlung (Exceptions) diskutiert.

**Prüfung**
schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten
**Beschreibung:**
Klausur zum Stoff des gesamten Moduls, insbesondere in Form von Programmieraufgaben. Bestehen der Klausur durch Erreichen von 50% der maximal erreichbaren Punkte.

| **Modul DSG-PKS-B Programmierung komplexer interagierender Systeme**<br>*Introduction to Parallel and Distributed Programming* | | 3 ECTS / 90 h<br>23 h Präsenzzeit<br>67 h Selbststudium |
|---|---|---|
| (seit WS17/18)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | | |
| **Inhalte:**<br>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<br>• Prozesse und Threads,<br>• Prozesskommunikation,<br>• Synchronisation bei Shared Memory,<br>• einfache C/S-Systeme mit TCP sockets,<br>• Message-Passing im Aktor-Modell.<br>Zusätzlich wird die Problematik robuster verteilter Systeme diskutiert und ein Ausblick auf alternative Interaktionsparadigmen gegeben. | | |
| **Lernziele/Kompetenzen:**<br>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. | | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand von 90 Std. gliedert sich in<br>• 22.5 Std. Teilnahme an der Praktischen Übung<br>• 55 Std. Bearbeiten der Programmieraufgabe (Assignment)<br>• 12 Std. Vorbereitung auf das Abschlusskolloquium<br>• 0.5 Std. Abschlusskolloquium inklusive Warten auf Ergebnis usw. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Betriebssysteme, wie sie z.B. im Modul PSI-EiRBS-B (vormals DSG-EiRBS-B) vermittelt werden.<br>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen<br>Modul Einführung in Rechner- und Betriebssysteme (PSI-EiRBS-B, vormals DSG-EiRBS-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **Programmierung komplexer interagierender Systeme**<br>**Lehrformen:** Übung<br>**Dozenten:** Mitarbeiter Praktische Informatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Lernziele:**<br>vgl. Modulbeschreibung<br>**Inhalte:**<br>vgl. Modulbeschreibung<br>**Literatur:**<br>- wird jeweils aktuell zur Veranstaltung angegeben - | **2,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Beschreibung:**<br>Die zu Beginn des Semesters ausgegebene Programmieraufgabe (Assignment) wird 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 der Programmieraufgabe vor, erläutern diese und beantworten Fragen zu ihrer Lösung und den dabei verwendeten Techniken und Sprachkonstrukten. | |

| **Modul DSG-Project-B Bachelorprojekt zur Praktischen Informatik**<br>*Bachelor Project in Distributed Systems* | 6 ECTS / 180 h<br>130 h Präsenzzeit<br>50 h Selbststudium |
|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | |
| **Inhalte:**<br>Überschaubare Themen aus der aktuellen Forschungsarbeit der Arbeitsgruppe Verteilte Systeme (DSG), die ohne umfangreiche Einarbeitung zu bearbeiten sind, werden in einer zum Teil gemeinsam, zum Teil arbeitsteilig, arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung im Rahmen eines 6-wöchigen 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. 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. Transformationen zwischen verschiedenen Prozesssprachen oder XML-Darstellungen, die Erstellung einfacher, neuer Werkzeuge im Kontext der Beschreibung und Analyse verteilter Systeme oder aber die Erweiterung von Werkzeugen um neue Funktionalitäten inklusive Einarbeitung in schon vorhandene Programmsysteme usw. Dabei wird sowohl durch die konkrete Themenstellung als auch die enge betreuung und Unterstützung des Projekts darauf geachtet, dass die gestellten Aufgaben auch im (fortgeschrittenen) Bachelorstudium sinnvoll zu bearbeiten sind. | |
| **Lernziele/Kompetenzen:**<br>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 Forschungsbreich der praktischen Informatik 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. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand von insgesamt 180 Std. gliedert sich in etwa in:<br>• 50 Std. Einführung, Vorstellen von Werkzeugen, Vorträge zum Projektstand<br>• 30 Std. Recherchen zu und Einarbeitung in Thematik des Praktikums inkl. Vorbereitung von Kurzvorträgen<br>• 80 Std. praktische Projektarbeit (Softwareentwicklung)<br>• 10 Std. Abfassen des Projektberichts<br>• 10 Std. Vorbereitung auf und Zeit für das Kolloquium (unter o.g. schon erbrachten Aufwänden) | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Programmierkenntnisse sowie Kenntnisse in den Grundlagen des im Projekt behandelten Themengebiets.<br>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| | | |
|---|---|---|
| Modul Einführung in Rechner- und Betriebssysteme (PSI-EiRBS-B, vormals DSG-EiRBS-B) - empfohlen | | |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** 1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Projektübung zur Praktischen Informatik**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>vgl. Modulbeschreibung<br>**Literatur:**<br>- je nach Projektthematik - | **4,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten<br>Bearbeitungsfrist: 2 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Anfertigen eines schriftlichen Berichts über das im Projekt durchgeführte Softwareprojekt. Diskussion des vorliegenden Projektberichts sowie der erstellten Artefakte vor dem Hintergrund des allgemeinen Themas der Projektarbeit. | |

| **Modul DSG-Sem-B Bachelorseminar zur Praktischen Informatik**<br>*Bachelor Seminar in Practical Computer Science* | | 3 ECTS / 90 h |
|---|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Guido Wirtz | | |
| **Inhalte:**<br>Verschiedene Themen aus dem Bereich der praktischen Informatik, die einen der fachlichen oder methodischen Aspekte aus den grundlegenden Informatik-Modulen DSG-EiAPS-B oder PSI-EiRBS-B anhand aktueller Literatur vertiefen und/oder ergänzen. | | |
| **Lernziele/Kompetenzen:**<br>Studierende sollen überschaubare aktuelle Themen der praktischen Informatik anhand eigener Literaturrecherchen unter Anleitung erarbeiten und in einer dem Thema angemessenen und für alle SeminarteilnehmerInnen verständlichen Form aufbereiten und präsentieren können. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Grundlegende Kenntnisse im jeweils im Seminar behandelten Gebiet der Praktischen Informatik, also mindestens eines der beiden Module DSG-EiAPS-B oder PSI-EiRBS-B.<br>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen<br>Modul Einführung in Rechner- und Betriebssysteme (PSI-EiRBS-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>2. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar zur Praktischen Informatik**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS<br>**Inhalte:**<br>vgl. Modulbeschreibung<br>**Literatur:**<br>- wird jeweils nach Seminarthemen vergeben - | **2,00 SWS** |

| **Prüfung** | |
|---|---|
| Referat mit schriftl. Hausarbeit / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:** | |

| | |
|---|---|
| Begutachtung einer schriftlichen Ausarbeitung zu den wichtigsten Aspekten des erarbeiteten Themas mit formgerechter Liste der verwendeten Literatur.<br>Teilnahme am Peer-Review der Ausarbeitungen anderer Teilnehmer*innen;<br>Freies Halten eines Referats auf der Grundlage der von dem/der Vortragenden erstellten Folien oder elektronischen Präsentationsunterlagen inklusive Diskussion der Inhalte mit den Seminarteilnehmerinnen und Seminarteilnehmern. | |

| **Modul DT-CPP-B Einführung in die Systemprogrammierung in C++**<br>*Introduction into Systems Programming in C++* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25 bis SS24)<br>Modulverantwortliche/r: Prof. Dr. Maximilian Schüle | | |
| **Inhalte:**<br>Vertiefung der Kenntnisse über moderne C++-Programmiertechniken und das C++-Ökosystem, Erlernen des Schreibens von gutem C++, Erlernen der Implementierung großer Systeme mit C++, Erlernen des Schreibens von Hochleistungscode mit C++. Wir erwarten nicht, dass Sie bereits Erfahrung in der Programmierung mit C oder C++ haben, aber Sie sollten mit einer allgemeinen Programmiersprache Ihrer Wahl vertraut sein. Außerdem sollten Sie mit gängigen Algorithmen und Datenstrukturen sowie mit Computerarchitektur und Betriebssystemen vertraut sein. | | |
| **Lernziele/Kompetenzen:**<br>Systemprogrammierung in C++ | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 3. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Einführung in die Systemprogrammierung in C++**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Maximilian Schüle<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Inhalte:**<br>Vertiefung der Kenntnisse über moderne C++-Programmiertechniken und das C++-Ökosystem, Erlernen des Schreibens von gutem C++, Erlernen der Implementierung großer Systeme mit C++, Erlernen des Schreibens von Hochleistungscode mit C++. Wir erwarten nicht, dass Sie bereits Erfahrung in der Programmierung mit C oder C++ haben, aber Sie sollten mit einer allgemeinen Programmiersprache Ihrer Wahl vertraut sein. Außerdem sollten Sie mit gängigen Algorithmen und Datenstrukturen sowie mit Computerarchitektur und Betriebssystemen vertraut sein. | **4,00 SWS** |
| **Prüfung**<br>Portfolio / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate | |

| | | |
|---|---|---|
| **Modul DT-DB4MLKD-B Moderne Datenbanksysteme für maschinelles Lernen und Wissensentdeckung**<br>*Modern Database Systems for Machine Learning and Knowledge Discovery* | | 3 ECTS / 90 h |
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Maximilian Schüle | | |
| **Inhalte:**<br>In this seminar, we study the challenges of modern database systems. We discuss the topic along with very recent publications about database systems for machine learning and knowledge discovery. | | |
| **Lernziele/Kompetenzen:**<br>selbständig Publikationen schreiben | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** jährlich nach Bedarf WS und SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Moderne Datenbanksysteme für maschinelles Lernen und Wissensentdeckung**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Maximilian Schüle<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** jährlich nach Bedarf WS und SS<br>**Lernziele:**<br>selbständig Publikationen schreiben<br>**Inhalte:**<br>In this seminar, we study the challenges of modern database systems. We discuss the topic along with very recent publications about database systems for machine learning and knowledge discovery. | **3,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 14 Tage | |

| | | |
|---|---|---|
| **Modul DT-Proj-B Bachelor-Projekt: Data Engineering**<br>*Bachelor Project: Data Engineering* | | 6 ECTS / 180 h |
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Maximilian Schüle | | |
| **Inhalte:**<br>Projekte zu Data Engineering | | |
| **Lernziele/Kompetenzen:**<br>vertiefendes wissenschaftliches Arbeiten im Rahmen eines Data Engineering Projektes | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| | |
|---|---|
| **Lehrveranstaltungen** | |
| **Bachelor-Projekt: Data Engineering**<br>**Lehrformen:** Projekt<br>**Dozenten:** Prof. Dr. Maximilian Schüle<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, SS<br>**Inhalte:**<br>Bachelor-Projekte zu Data Engineering | **4,00 SWS** |
| **Prüfung**<br>Kolloquium, schr. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 3 Monate | |

| **Modul GdI-GTI-B Grundlagen der Theoretischen Informatik**<br>*Machines and Languages* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | | |
| **Inhalte:**<br>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 steht 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 besprochen, insbesondere die Chomsky Hierarchie und Komplexitätsklassen wie P, NP, PSPACE, EXPTIME und ihre Beziehung untereinander. | | |
| **Lernziele/Kompetenzen:**<br>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. | | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Vorlesung und Übung: 45 Stunden<br>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden<br>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden<br>• Prüfungsvorbereitung + schriftliche Prüfung (90 min.): 30 Stunden | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>gute Englischkenntnisse<br>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) -<br>Modul Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (GdI-MfI-1) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **1. Grundlagen der Theoretischen Informatik**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Ph.D. Michael Mendler<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.<br>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.<br>**Literatur:**<br>• Hopcroft, J. E., Motwani, R., Ullman, J. D.: Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, Pearson Studium, 2002.<br>• Asteroth, A., Baier, Ch.: Theoretische Informatik, Pearson Studium, 2002.<br>• Martin, J. C.: Introduction to Languages and the Theory of Computation, McGraw Hill, (2nd ed.), 1997. | **2,00 SWS** |
| **2. Grundlagen der Theoretischen Informatik**<br>**Lehrformen:** Übung<br>**Dozenten:** Prof. Ph.D. Michael Mendler, N.N.<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.<br>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben. | **2,00 SWS** |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>Die Prüfungssprache ist Englisch. | |

| **Modul GdI-MTL-B Modal and Temporal Logic**<br>*Modal and Temporal Logic* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | |
| **Inhalte:**<br>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 *Computational Logic* 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:**<br>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. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Vorlesung und Übung: 45 Stunden<br>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden<br>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden<br>• Prüfungsvorbereitung: 30 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>grundlegende Programmierkenntnisse, gute Englischkenntnisse<br>Modul Diskrete Modellierung (Inf-DM-B) - empfohlen<br>Modul Logik und Berechenbarkeit (Inf-LBR-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Modal and Temporal Logic**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Ph.D. Michael Mendler<br>**Sprache:** Englisch/Deutsch | **4,00 SWS** |

**Angebotshäufigkeit:** WS, jährlich

**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.

**Literatur:**


- 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.


**Prüfung**

mündliche Prüfung

**Beschreibung:**

Die Prüfungssprache ist Englisch.

Die Prüfung wird in Abhängigkeit von der Anzahl der Teilnehmer als mündliche Prüfung (30 Minuten) oder als schrifliche Prüfung (90 Minuten) durchgeführt. Die Prüfungsform wird den Teilnehmern am Anfang des Semesters bekanntgegeben.

| **Modul GdI-MfI-1 Mathematik für Informatik 1 (Aussagen- u. Prädikatenlogik)**<br>*Propositional and Predicate Logic* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | |
| **Inhalte:**<br>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:**<br>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. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Vorlesung und Übung: 45 Stunden<br>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 60 Stunden<br>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 45 Stunden<br>• Prüfungsvorbereitung + schriftliche Prüfung (90 min.): 30 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>gute Englischkenntnisse | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **1. Mathematik für Informatik 1**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Ph.D. Michael Mendler<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |
| **Inhalte:** | |

| | |
|---|---|
| In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.<br><br>**Literatur:**<br>• Ehrig, H., Mahr, B., Cornelius, F., Große-Rhode, Zeitz, M. P.: Mathematisch strukturelle Grundlagen der Informatik. Springer Verlag, 2. Aufl., 2001.<br>• Grassmann, W. K., Tremblay, J.-P.: Logic and Discrete Mathematics - A Computer Science Perspective. Prentice Hall, 1996.<br>• Scheinerman, E. R.: Mathematics – A Discrete Introduction. Brooks/Cole, 2000.<br>• Barwise, J., Etchemendy, J: Language, Proof, and Logic. Seven Bridges Press, 2000. | |
| **2. Mathematik für Informatik 1**<br>**Lehrformen:** Übung<br>**Dozenten:** Prof. Ph.D. Michael Mendler, N.N.<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, jährlich<br><br>**Inhalte:**<br>Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung. | **2,00 SWS** |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten | |

| **Modul GdI-Proj-B Bachelorprojekt Grundlagen der Informatik**<br>*Foundations of Computing Project* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | | |
| **Inhalte:**<br>Im Projektmodul werden wechselnde Themen angeboten, etwa zum Einsatz automatischer Verifikationswerkzeuge (Theorembeweiser, Modellprüfer, Verzögerungsanalyse) oder zum Bau und der Anwendung von visuellen Entwurfswerkzeugen für eingebettete Systeme (UML, Statecharts). Ein weiterer Bereich ist die prototypische Implementierung neuer algorithmischer Verfahren aus aktuellen Forschungsgebieten der Arbeitsgruppe (Informationssicherheit, Theorie verteilter Systeme, Logik). | | |
| **Lernziele/Kompetenzen:**<br>Fähigkeit zur selbständigen Erarbeitung von Problemlösungen, sowohl auf der Basis des erlernten Wissens und der angeeigneten Fähigkeiten aus dem Studium als auch der aktuellen wissenschaftlichen Literatur; Fähigkeit, komplexe Problemlösungsansätze im Rahmen eines systematischen ingenieurtechnischen Entwicklungsprozesses in Software umzusetzen und professionell zu dokumentieren; Fähigkeit zur Teamarbeit; Wissenschaftliche Neugier und die Ausbildung einer selbstbewussten und forschenden Einstellung zur Technik. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>grundlegende Programmierkenntnisse, insbesondere in funktionaler Programmierung; gute Englischkenntnisse<br><br>Modul Modal and Temporal Logic (GdI-MTL-B) - empfohlen<br>Modul Diskrete Modellierung (Inf-DM-B) - empfohlen<br>Modul Logik und Berechenbarkeit (Inf-LBR-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **GdI Projekt**<br>**Lehrformen:** Projektseminar<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS | **4,00 SWS** |
| **Inhalte:**<br>In der Projektübung werden wechselnde Themen angeboten, etwa zum Einsatz automatischer Verifikationswerkzeuge (Theorembeweiser, Modellprüfer, Verzögerungsanalyse) oder zum Bau und der Anwendung von visuellen Entwurfswerkzeugen für eingebettete Systeme (UML, Statecharts). Ein weiterer Bereich ist die prototypische Implementierung neuer algorithmischer Verfahren aus aktuellen Forschungsgebieten der Arbeitsgruppe (Informationssicherheit, Theorie verteilter Systeme, Logik). Die Aufgabenstellung wird bei Ankündigung bzw. zu Beginn des Projektes bekanntgegeben.<br><br>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben. | |

| | |
|---|---|
| **Literatur:**<br>Literatur wird bei Ankündigung bzw. zu Beginn des Projektes bekanntgegeben. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Darstellung der Projektergebnisse in einer Hausarbeit und deren Verteidigung in einem Kolloquium.<br>Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben. | |

| **Modul GdI-Sem-B Bachelorseminar Grundlagen der Informatik**<br>*Seminar Foundations of Computing* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS17/18)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | | |
| **Inhalte:**<br>Im Seminarmodul werden wechselnde Themen im Bereich der Informatikgrundlagen angeboten. | | |
| **Lernziele/Kompetenzen:**<br>Fähigkeit zur selbständigen Erarbeitung von Inhalten aus der aktuellen wissenschaftlichen Literatur; Fähigkeit, komplexe Problemlösungsansätze schriftlich und mündlich zu vermitteln. Förderung der wissenschaftliche Neugier und die Ausbildung einer selbstbewussten und forschenden Einstellung zur Technik. | | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Seminarvorträgen: 15 Stunden<br>• Recherche und Literaturstudium: 25 Stunden<br>• Vorbereitung des Seminarvortrags und schriftliche Ausarbeitung: 50 Stunden | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Mathematik für Informatiker, Einführung in die Informatik, Rechner- und Betriebssysteme, Grundlagen der Theoretischen Informatik, gute Englischkenntnisse. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** jährlich nach Bedarf WS und SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Grundlagen der Informatik**<br>**Lehrformen:** Seminar<br>**Dozenten:** Michael Mendler, N.N.<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** jährlich nach Bedarf WS und SS<br>**Inhalte:**<br>Das GdI-Seminar wird zu semesterweise wechselnden Themen angeboten.<br>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.<br>**Literatur:**<br>Literatur wird bei Ankündigung bzw. zu Beginn des Seminars bekanntgegeben. | **2,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:** | |

| | |
|---|---|
| Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben. | |

| **Modul HCI-IS-B Interaktive Systeme**<br>*Interactive Systems* | | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|---|
| (seit SS23)<br>Modulverantwortliche/r: Prof. Dr. Tom Gross | | |
| **Inhalte:**<br>Theoretische, methodische und praktische Grundlagen der Mensch-Computer-Interaktion. | | |
| **Lernziele/Kompetenzen:**<br>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. | | |
| **Sonstige Informationen:**<br>http://www.uni-bamberg.de/hci/leistungen/studium<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden<br>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden<br>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden<br>• Bearbeiten der optionalen Studienleistungen: insgesamt ca. 45 Stunden<br>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)<br>Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Grundkenntnisse in Informatik im Umfang einer Einführung in die Informatik | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Interaktive Systeme**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Tom Gross<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |
| **Inhalte:**<br>Im Rahmen der Vorlesung werden die folgenden Themen behandelt:<br>• Einführung in die Gestaltung von Benutzungsoberflächen<br>• Benutzer und Humanfaktoren<br>• Maschinen und technische Faktoren<br>• Interaktion, Entwurf, Prototyping und Entwicklung | |

| | |
|---|---|
| • Evaluierung von interaktiven Systemen<br>• Entwicklungsprozess interaktiver Systeme<br>• Interaktive Systeme im größeren Kontext und verwandte Themen<br><br>**Literatur:**<br>Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:<br>• Sharp, H., Rogers, Y. and Preece, J. Interaction Design: Beyond Human-Computer Interaction. (5th ed.). Wiley, New York, 2019<br>• Dix, A., Finlay, J., Abowd, G.D. und Beale, R. Human-Computer Interaction. Pearson, Englewood Cliffs, NJ, 3. Auflage, 2004. | |
| **2. Interaktive Systeme**<br>**Lehrformen:** Übung<br>**Dozenten:** Mitarbeiter Mensch-Computer-Interaktion<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br><br>**Inhalte:**<br>praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen<br><br>**Literatur:**<br>siehe Vorlesung | **2,00 SWS** |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>In Abhängigkeit der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur oder in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.<br><br>In der Klausur über 90 min. können 90 Punkte erzielt werden.<br><br>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 Prüfung 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar. | |

| | |
|---|---|
| **Prüfung**<br>mündliche Prüfung<br>**Beschreibung:**<br>In Abhängigkeit der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur oder in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben. | |

| | |
|---|---|
| In der mündlichen Prüfung können 90 Punkte erzielt werden. Die Prüfungsdauer wird im ersten Veranstaltungstermin mitgeteilt.<br><br>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 Prüfung 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar. | |

| **Modul HCI-Sem-B Bachelorseminar Mensch-Computer-Interaktion**<br>*Bachelor-Seminar Human-Computer Interaction* | 3 ECTS / 90 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Tom Gross | |
| **Inhalte:**<br>Aktive wissenschaftliche Bearbeitung aktueller Konzepte, Technologien und Werkzeuge der Mensch-Computer-Interaktion. | |
| **Lernziele/Kompetenzen:**<br>Ziel ist das Erlernen des eigenständigen Erarbeitens und Präsentierens von Themengebieten aus dem Fach Mensch-Computer-Interaktion auf Basis der Literatur. Dabei werden die Fähigkeiten im Bereich der kritischen und systematischen Literaturbetrachtung ebenso weiterentwickelt wie die Kompetenzen in der Präsentation von Fachthemen. | |
| **Sonstige Informationen:**<br>http://www.uni-bamberg.de/hci/leistungen/studium<br>Der Arbeitsaufwand für dieses Modul gliedert sich typischerweise in folgende Bereiche:<br>• Teilnahme an den Präsenzveranstaltungen (Themenvergabe, Besprechungen, Präsentationen): ca. 20 Stunden<br>• Literaturrecherche und Einarbeitung: ca. 25 Stunden<br>• Vorbereitung der Präsentation: ca. 15 Stunden<br>• Erstellen der schriftlichen Ausarbeitung: ca. 30 Stunden<br>Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Modul Interaktive Systeme (HCI-IS-B) | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Mensch-Computer-Interaktion**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Tom Gross, Mitarbeiter Mensch-Computer-Interaktion<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Inhalte:**<br>Im Seminar werden aufbauend auf den in den Vorlesungen und Übungen des Faches Mensch-Computer-Interaktion erworbenen Kenntnissen und Fertigkeiten wechselnde aktuelle Forschungsthemen zu deren Inhalten bearbeitet. Dabei sind im Regelfall Aspekte mehrerer Lehrveranstaltungen relevant. | |
| **Literatur:** | |

| | |
|---|---|
| wird jeweils zu Beginn der Veranstaltung bekannt gegeben | |
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Beschreibung:**<br>Schriftliche Ausarbeitung und Vortrag zu dem im Seminar von der Teilnehmerin bzw. vom Teilnehmer bearbeiteten Thema, inkl. Diskussion | |

| **Modul HCI-US-B Ubiquitäre Systeme**<br>*Ubiquitous Systems* | 6 ECTS / 180 h |
|---|---|

(seit WS24/25)
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.

**Sonstige Informationen:**
http://www.uni-bamberg.de/hci/leistungen/studium

Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:

- 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 optiopnalen Studienleistungen): ca. 75 Stunden
- Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)

Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:** | **Besondere Bestehensvoraussetzungen:** |
|---|---|
| Modul Algorithmen und Datenstrukturen (AI-AuD-B)<br>Modul Einführung in die Informatik (Inf-Einf-B) | keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** 1 Semester |
|---|---|---|

**Lehrveranstaltungen**

| **Ubiquitäre Systeme**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Tom Gross<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |
|---|---|
| **Inhalte:**<br>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: | |

- Grundlegende Konzepte
- Basistechnologie und Infrastrukturen
- Ubiquitäre Systeme und Prototypen
- Kontextadaptivität
- Benutzerinteraktion
- Ubiquitäre Systeme im größeren Kontext und verwandte Themen

**Literatur:**
Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:

- Krumm, J., (Hrsg.). Ubiquitous Computing Fundamentals. Taylor & Francis Group, Boca Raton, FL, 2010.

**Prüfung**
mündliche Prüfung
**Beschreibung:**
In Abhängigkeit der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur oder in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.

In der mündlichen Prüfung können 90 Punkte erzielt werden. Die Prüfungsdauer wird im ersten Veranstaltungstermin mitgeteilt.

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 Prüfung 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.

**Lehrveranstaltungen**

**Ubiquitäre Systeme** — **2,00 SWS**
**Lehrformen:** Übung
**Dozenten:** Mitarbeiter Mensch-Computer-Interaktion
**Sprache:** Deutsch/Englisch
**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**
praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen.

**Literatur:**
siehe Vorlesung

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

| | |
|---|---|
| In Abhängigkeit der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur oder in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.<br><br>In der Klausur über 90 min. können 90 Punkte erzielt werden.<br><br>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 Prüfung 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar. | |

| **Modul Inf-DM-B Diskrete Modellierung**<br>*Discrete Modeling* | 9 ECTS / 270 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Isolde Adler | |
| **Inhalte:**<br>Modellieren ist eine grundlegende Arbeitstechnik in vielen Bereichen der Informatik und darüber hinaus. Modelle dienen der exakten Beschreibung von Szenarien und sind damit die Voraussetzung zum Lösen von Problemen mittels Methoden der Informatik. Dabei ist es wichtig, die Modelle passend zur Problemstellung zu wählen. Hier bietet die diskrete Mathematik ein vielfältiges Handwerkszeug.<br>Für das nachhaltige, verlässliche Modellieren sowie für das Lösen von Problemen ist es wichtig, das exakte Argumentieren zu erlernen. Deshalb ist das Einüben der Sprache der Mathematik ein zentrales Thema in diesem Modul. Sie bietet die Sicherheit, sich auf die Modelle und Lösungen verlassen zu können.<br>In diesem Modul werden Aussagen- und Prädikatenlogik, Mengen, Relationen und Funktionen, Graphen, Bäume, und Methoden der Kombinatorik, formale Sprachen und endliche Automaten eingeführt und anhand von Modellierungsbeispielen besprochen. Zudem werden mathematische Beweistechniken eingeführt und eingeübt. | |
| **Lernziele/Kompetenzen:**<br>Vertrautheit mit unterschiedlichen Modellierungsmethoden<br>Sicherheit im mathematisch exakten Argumentieren<br>Vertrautheit mit grundlegenden Definitionen und Eigenschaften aus dem Bereich der diskreten Mathematik und mit deren Rolle in der Informatik<br>Sicherheit in der Entwicklung von Strategien zur Problemlösung<br>Analytische Fähigkeiten | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Keine | |
| **Empfohlene Vorkenntnisse:**<br>Interesse an formalen Methoden. Dies ist eine grundlegende Veranstaltung, die für die ersten Studiensemester empfohlen wird. | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich<br>**Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Diskrete Modellierung**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Isolde Adler<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **6,00 SWS** |
| **Inhalte:** | |

| | |
|---|---|
| In der Vorlesung werden die Themen motiviert und eingeführt, im Detail erklärt sowie Techniken und Methoden vorgestellt. Es werden Beispiele, Beweise, typische Fragestellungen und Anwendungen in der Informatik besprochen. | |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (Klausur) | |

| **Modul Inf-Einf-B Einführung in die Informatik**<br>*Introduction to Computer Science* | 9 ECTS / 270 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Dominik Herrmann | |

**Inhalte:**

In diesem Modul erhalten die Studierenden eine umfassende Einführung in die Informatik. Dazu werden gängige Prinzipien der Programmierung und Techniken zur Problemlösung, sowohl mit als auch ohne Code, vermittelt. Dies befähigt die Studierenden, sich eigenständig in Programmiersprachen einzuarbeiten und komplexe Problemstellungen zu bearbeiten.

Nach einer Einführung essenzieller Konzepte wie Variablen, Funktionen, Bedingungen und Schleifen machen sich die Studierenden mit gängigen Linux-Kommandozeilenprogrammen und Dateisystemkonzepten vertraut, die eine text- und dateibasierte Datenverarbeitung mittels Shell-Skripten ermöglichen. Dies bildet die Basis für die Einführung in die systemnahe Programmiersprache C. Dabei werden imperative und prozedurale Programmierung sowie dynamische Speicherverwaltung, stapelbasierte Programmausführung und für die Programmierung relevante Mechanismen der Datenrepräsentation vermittelt (bspw. Overflows, Unicode, Escape-Sequenzen). Im weiteren Verlauf wird die Programmiersprache Python eingeführt, anhand derer Konzepte moderner Programmiersprachen sowie objektorientierter und funktionaler Programmierparadigmen erörtert werden. Parallel dazu werden Arbeitstechniken zur Erstellung nachvollziehbarer und sicherer Programme vermittelt, etwa Debugging und Quellcodedokumentation, und verbreitete Fehlertypen aufgezeigt.

Das Modul bietet zudem einen ersten Einblick in Algorithmen, abstrakte Datentypen und gängige Datenstrukturen. Die Studierenden implementieren iterative und rekursive Algorithmen zur Lösung grundlegender Probleme wie Sortieren und Suchen und modellieren Problemlösungen mit passenden Datenstrukturen wie Listen, Bäumen, Tries und Hash-Tabellen. Ergänzend erhalten sie Einblicke in die für die Anwendungsprogrammierung relevanten Aspekte von Rechnernetzen (TCP/IP). Abschließend erhalten die Studierenden einen Einblick in die Paradigmen, die bei der Entwicklung einfacher Webanwendungen mit HTML, Python und JavaScript zum Einsatz kommen.

Die Inhalte werden theoretisch fundiert; der Schwerpunkt des Moduls liegt jedoch auf der Entwicklung praktischer Problemlösungskompetenzen durch Übungsaufgaben, die ein intensives Selbststudium erfordern. Das Erlernte wird durch ein semesterbegleitendes Programmierprojekt, das am Ende des Semesters präsentiert wird, angewandt und gefestigt.

**Lernziele/Kompetenzen:**

Nach Abschluss des Moduls sind die Studierenden in der Lage, prozedural sowie mit objektorientierten und funktionalen Konzepten zu programmieren und dabei die im Modul vermittelten grundlegenden Programmierkonzepte und -techniken anzuwenden. Darüber hinaus können die Studierenden Informationen angemessen darstellen und verarbeiten, Probleme analysieren, Muster in Problemstellungen erkennen und komplexe Problemstellungen in kleinere, handhabbare Teile zerlegen. Sie verstehen die verschiedenen Abstraktionsebenen eines Programms und können zwischen Entwurfs- und Implementierungsdetails unterscheiden.

Die Studierenden sind auch in der Lage, Korrektheit, Design und Stil von Code zu beurteilen und neue Programmiersprachen selbständig zu erlernen. Sie können eine vorhandene Lösung testen, Fehler und ausgewählte Sicherheitsprobleme identifizieren und Spezialfälle in der Problemstellung erkennen. Darüber hinaus können sie Situationen erkennen, in denen ein Trade-off zwischen Laufzeit und Speicherplatz besteht. Mit den Grundlagen von Rechnernetzen und Webanwendungen sind sie vertraut. Schließlich

können sie ihre Programmierkenntnisse anwenden, um ein selbst gewähltes Problem zu lösen und ihre Lösung zu präsentieren.

**Sonstige Informationen:**
Der Arbeitsaufwand von 270 Stunden verteilt sich ausgehend von einem 15 Arbeitswochen dauernden Semester in etwa wie folgt:
30 Std. Vorlesungsteilnahme in Präsenz
30 Std. Übungsteilnahme in Präsenz
60 Std. Bearbeiten von wöchentlichen Übungsaufgaben, d.h. ca. 4 Std./Woche
90 Std. Vor- und Nachbereitung von Vorlesung und Übung, d.h. ca. 6 Std./Woche
40 Std. Programmierprojekt
20 Std. Vorbereitung auf und Zeit für die Abschlussklausur
Bei diesem Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:** keine | | **Besondere Bestehensvoraussetzungen:** keine |
|---|---|---|
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** ab dem 1. | **Minimale Dauer des Moduls:** Semester |

**Lehrveranstaltungen**

**1. Vorlesung** — **4,00 SWS**
**Lehrformen:** Vorlesung
**Sprache:** Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Literatur:**
Für diesen Kurs werden keine Bücher benötigt oder empfohlen. Die unten aufgeführten Bücher könnten jedoch von Interesse sein.

- Hacker's Delight, Zweite Ausgabe, Henry S. Warren Jr., Pearson Education, 2013
- How Computers Work, Zehnte Ausgabe, Ron White, Que Publishing, 2014
- Programmieren in C, Vierte Ausgabe, Stephen G. Kochan, Pearson Education, 2015

**2. Übung** — **2,00 SWS**
**Lehrformen:** Übung
**Sprache:** Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**
In der Übung werden die wichtigsten Konzepte der Vorlesung an theoretischen und praktischen Beispielen veranschaulicht und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema vertieft.

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 180 Minuten<br>**Beschreibung:**<br>Die maximale Punktzahl, die in der Prüfung erreicht werden kann, beträgt 100. Teilnehmer, die eine semesterbegleitende Studienleistung erbringen können bis zu 10 Bonuspunkte erreichen. Details zu Anforderungen und Abgabefristen werden in der ersten Vorlesung bekannt gegeben. Wenn die in der Prüfung erreichte Punktzahl ausreicht, um damit allein die Prüfung zu bestehen (in der Regel ist dies der Fall, wenn mindestens 50 von 100 Punkten erreicht wurden), werden die Bonuspunkte zu den in der Prüfung erreichten Punkten addiert. Die Note 1,0 kann auch ohne die Bonuspunkte erreicht werden. | |

| **Modul Inf-GRABS-B Grundlagen der Rechnerarchitektur und Betriebssysteme**<br>*Foundations of Computer Architecture and Operating Systems* | | 9 ECTS / 270 h |
|---|---|---|
| (seit SS25)<br>Modulverantwortliche/r: Prof. Dr. Michael Engel | | |
| **Inhalte:**<br>Das Modul behandelt die Grundlagen der Hardware und Systemsoftware moderner Computersysteme mit einem Schwerpunkt auf der Interaktion zwischen Soft- und Hardware und deren Auswirkungen auf Systemeigenschaften wie Performanz, Energieaufnahme, Sicherheit und Zuverlässigkeit. Basierend auf einer Einführung in die technischen Grundlagen der Informatik wie digitale Schaltungen, Architektur von Prozessoren, Speicherhierarchie, Ein-/Ausgabeeinheiten und Bussystemen sowie Informationsdarstellung (Zahlenformate, Zeichencodierung), digitale Logik und Arithmetik wird die Nutzung, Verwaltung und Zuteilung der Hardwarekomponenten durch Systemsoftware, insbesondere Betriebssysteme, erläutert.<br>Hierbei sind wichtige Schwerpunkte das Zusammenspiel von Software und Hardware, die Realisierung und Steuerung von Nebenläufigkeit und Parallelität in Rechnersystemen sowie die Kommunikation, Synchronisation und Isolation verschiedener nebenläufiger Aktivitäten sowie die Verwaltung und Optimierung von Zugriffen auf die verschiedenen Elemente der Speicherhierarchie.<br>Die Themen werden anhand der Prozessorarchitektur RISC-V, systemnahen Programmiersprachen (C und Assembler) und Beispielen moderner Betriebssysteme (z.B. Linux) behandelt. Ergänzend wird ein erster Einblick in Rechnernetze und Aspekte der Systemsicherheit gegeben.<br>Im Rahmen der Veranstaltung erarbeiten sich die Studierenden zusätzlich praktische Kenntnisse im Umgang mit der Unix-Kommandozeile sowie der Assembler-Programmierung. Diese Inhalte erarbeiten sich die Studierenden auch anhand von bereitgestellten Materialien und Aufgaben primär im Selbststudium. | | |
| **Lernziele/Kompetenzen:**<br>Studierende erhalten einen umfassenden Einblick in die systemnahen Bereiche der Informatik und verstehen die Interaktion von Software und Hardware und die Auswirkungen von Hardwareeigenschaften auf nichtfunktionale Eigenschaften von Software wie Performanz oder Energieaufnahme. Die Studierenden können den Aufbau und die grundlegende Funktionalität moderner Prozessoren, Rechnersysteme und Betriebssysteme erläutern und haben ein grundlegendes Verständnis nebenläufiger und paralleler Prozesse und der zugehörigen Methoden zur Kommunikation, Synchronisation und Isolation. Die Studierenden sind dazu in der Lage, Assemblerprogramme zu entwerfen und den Zusammenhang von Hochsprachen- zu Assemblerbefehlssequenzen sowie die zugehörigen Vorgänge auf Seiten der Hardware zu bestimmen. Zusätzlich können die Studierenden die Unix-Befehlszeile zur Entwicklung von Software einsetzen. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Grundlegende Programmierkenntnisse, z.B. aus EIAPS oder der entsprechenden Nachfolgeveranstaltung | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 2. | **Minimale Dauer des Moduls:**<br>1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **1. Vorlesung Grundlagen der Rechnerarchitektur und Betriebssysteme**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Michael Engel<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>siehe Modulbeschreibung<br>**Inhalte:**<br>siehe Modulbeschreibung<br>**Literatur:**<br>Frank Slomka, Michael Glaß<br>**Grundlagen der Rechnerarchitektur: Von der Schaltung zum Prozessor**<br>Springer Fachmedien Wiesbaden GmbH, 1. Auflage 2023<br>ISBN-13: 978-3-658-36658-2 (Softcover) / 978-3-658-36659-9 (eBook)<br>Andrew Waterman, David A. Patterson<br>**The RISC-V Reader: An Open Architecture Atlas**<br>Strawberry Canyon, 2017. ISBN-13: 978-0-999-24911-6<br>Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau<br>**Operating Systems: Three Easy Pieces**<br>https://pages.cs.wisc.edu/~remzi/OSTEP/<br>Weitere Literatur wird nach Bedarf zur Verfügung gestellt. | **4,00 SWS** |
| **2. Übung Grundlagen der Rechnerarchitektur und Betriebssysteme**<br>**Lehrformen:** Praktikum, Übung<br>**Dozenten:** Prof. Dr. Michael Engel<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>siehe Modulbeschreibung<br>**Inhalte:**<br>siehe Modulbeschreibung | **2,00 SWS** |

| Prüfung | |
|---|---|
| schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 90 Minuten<br>Bearbeitungsfrist: 90 Minuten | |

| **Modul Inf-LBR-B Logik und Berechenbarkeit**<br>*Logic and Computability* | | 9 ECTS / 270 h |
|---|---|---|
| (seit SS25)<br>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler | | |
| **Inhalte:**<br>Die Grundlagenvorlesung vermittelt die Kenntnis elementarer Konstruktionen und Methodiken der Logik und Berechenbarkeitstheorie sowie die Fähigkeit, diese an Beispielen anzuwenden. Die Veranstaltung bietet darüber hinaus in diesen Themengebieten eine Einführung in zentrale theoriebildende Ergebnisse der Informatik als eigenständige wissenschaftliche Disziplin. | | |
| **Lernziele/Kompetenzen:**<br>Lernziele sind die Kenntnis elementarer Grundbegriffe der Beweis- und Modelltheorie der klassischen Aussagen- und Prädikatenlogik; Die Kenntnis der grundlegenden Definitionen der Komplexitätstheorie zur Erfassung der Ausdruckskraft und Leistungsfähigkeit von logischen Formalismen und algorithmischen Strukturen. Einsicht in die Grenzen der algorithmischen Entscheidbarkeit und Berechenbarkeit.<br><br>Durch die Veranstaltung soll insbesondere die Fähigkeit vermittelt werden, umgangssprachlich gegebene Strukturen und Prozesse der natürlichen und technischen Umwelt, speziell solcher von nicht-numerischer Natur, mit symbolischen Formalismen zu erfassen und mit Hilfe kombinatorischer, logischer und algorithmischer Lösungsansätze zu analysieren; Die Fähigkeit zur mathematischen Abstraktion; 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. | | |
| **Sonstige Informationen:**<br>Die Veranstaltung besteht aus zwei Teilen unterschiedlichen Umfangs. Der erste Teil behandelt die Logik und der zweite Teil die Berechenbarkeitstheorie. Beide Teile werden sequenziell im selben Semester und aufeinander aufbauend angeboten.<br><br>In beiden Teilen wird jeweils in der Vorlesung das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben. Die angeschlossenen Übungen vertiefen die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen.<br><br>Der Arbeitsaufwand für dieses Modul gliedert sich insgesamt grob wie folgt:<br>• Teilnahme an Vorlesungen und Übungen: 67 Stunden<br>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden<br>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 65 Stunden<br>• Prüfungsvorbereitung + Teilnahme and Prüfungen: 48 Stunden | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>gute Englischkenntnisse<br><br>Modul Diskrete Modellierung (Inf-DM-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** |

| | | |
|---|---|---|
| | | 1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **Logik und Berechenbarkeit**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Ph.D. Michael Mendler, N.N.<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich | **6,00 SWS** |
| **Lernziele:**<br>Teil1:<br>Kenntnis elementarer Konzepte der Mengenlehre mit Fokus auf Ordnungen und Verbände, sowie ihre Rolle für Induktion und Rekursion; Syntax der typisierten Prädikatenlogik erster Stufe (FOL); Fähigkeit zur Formalisierung natürlichsprachiger Spezifikationen in FOL; Kenntnisse zur Beweistheorie für FOL, insbesondere zum Kalkül des natürlichen Schließens und die Fähigkeit zur Formalisierung von logischen Argumentationsketten, insbesondere von Induktionsbeweisen. Kenntnis der Axiomatisierung wichtiger mathematischer Strukturen; Kenntnis und Fähigkeit zur Nutzung wichtiger Fragmente von FOL, speziell von Pränexen und Skolem Normalformen; Einsicht in die spezielle Natur von Hornformeln; Kenntnis der Tarskischen Semantik und ihrer Bedeutung für die mathematische Definition des Begriffs der Wahrheit; Korrektheit und Vollständigkeitssätze; Verständnis der Begriffe "Modell" und "Theorie" sowie Wissen über zentrale Ergebnisse zur Ausdruckskraft, insbesondere Unvollständigkeitssätze (Peano Arithmetik) und Sätze zur Kategorizität und Kompaktheit von FOL.<br>Teil 2:<br>Kenntnis des Konzepts der Turingmaschine als Basismodell der Berechenbarkeitstheorie und Fähigkeit, konkrete algorithmische Probleme im Turingmodell zu formalisieren; Verständnis des Unterschieds in der Komplexität von Berechnung, Algorithmus und Problem; Kenntnis der Rates-of-Growth Klassifikation und Fähigkeit zur Anwendung, Einsicht in die Abhängigkeit des Rates-of-Growth Klassifikation vom Maschinenmodell; Blum's Speedup Theorem; Verständnis für die Unterschiede von Berechenbarkeit, Aufzählbarkeit und Entscheidbarkeit; Fähigkeit, einfache algorithmische Problemstellungen hinsichtlich ihrer Berechenbarkeit einzuschätzen. Einsicht in die Bedeutung des Unterschieds zwischen deterministischen und nichtdeterministischen Berechnungsmodellen; Kenntnis von Ergebnissen zur Unentscheidbarkeit (insb. Halteproblem); Kenntnis wichtiger Äquivalenzen und Hierarchien von Komplexitätsklassen (insbesondere. PTIME, NP, co-NP); Reduktionen und Vollständigkeit; 3SAT und Cook's Theorem; Kenntnis der Komplexität wichtiger Beispielprobleme aus der Informatik (insbesondere Unentscheidbarkeit der Peanoarithmetik, Hilbert's 10th Problem; Entscheidbarkeit von Fragmenten der Prädikatenlogik).<br>In der Beschäftigung mit mathematischen Modellen der Berechenbarkeit sollen Kompetenzen vermittelt werden, um Berechnungsmodelle unterschiedlicher Ausdruckskraft systematisch aufeinander zu reduzieren und die Äquivalenz | |

von Rechenmodellen nachzuweisen oder zu widerlegen; Kenntnis konkreter mathematischer Grundmodelle zur Beschreibung von Algorithmus und Prozess, welche die wissenschaftlich-methodische Basis der Informatik bilden.

**Inhalte:**

Im ersten Teil der Veranstaltung werden die 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 ausführlich erläutert. Für den Formalismus der Prädikatenlogik erster Stufe werden Beweistechniken sowie wesentliche Ergebnisse zur Semantik und Ausdruckskraft besprochen.

Im zweiten Teil wird das Modell der Turingmaschine als das Standardmodell der Berechenbarkeit und historischer Ausgangspunkt für die Entwicklung programmierbarer Rechenmaschinen eingeführt und seine mathematischen Eigenschaften analysiert. Am Beispiel der Turingprogramme wird zunächst die formale Verifikation mittels logischer Invarianten eingeübt. Mit Turingmaschinen und anderer damit äquivalenter Berechnungsmodelle werden die wichtigsten grundlegenden Begriffe der Berechenbarkeits- und Komplexitätstheorie vermittelt. Insbesondere werden zentrale strukturelle Hierarchien der Komplexität vorgestellt und an ausgewählten Beispielen aus der Informatik besprochen. Durch das Studium der intrinsischen Grenzen des formalistischen und logizistischen Methode soll eine kritische Haltung im Verständnis von algorithmischer Berechenbarkeit gefördert werden.

**Literatur:**

Teil 1:


- J. Donald Monk: Mathematical Logic. Springer 1976.
- 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.
- Barwise, J., Etchemendy, J: Language, Proof, and Logic. Seven Bridges Press, 2000.


Teil 2:


- Hopcroft, J. E., Motwani, R., Ullman, J. D.: Introduction to Automata Theory, Languages, and Computation. Addison Wesley, 2001.
- J. Donald Monk: Mathematical Logic, Springer 1976.
- Martin, J. C.: Introduction to Languages and the Theory of Computation, McGraw Hill, (2nd ed.), 1997.
- Sudkamp, Th. A.: Languages and Machines. An Introduction to the Theory of Computer Science. Addison Wesley, (2nd ed.) 1997.


**Prüfung**

| | |
|---|---|
| schriftliche Prüfung (Klausur) / Prüfungsdauer: 135 Minuten | |

| **Modul KInf-Seminar-B Bachelorseminar Kulturinformatik**<br>*Bachelorseminar Kulturinformatik* | 3 ECTS / 90 h<br>23 h Präsenzzeit<br>67 h Selbststudium |
|---|---|
| (seit SS20)<br>Modulverantwortliche/r: Prof. Dr. Christoph Schlieder | |
| **Inhalte:**<br>Das Modul führt anhand der Forschungsliteratur in grundlegende Methoden aus dem Bereich der Kulturinformatik ein. Die behandelten Problemstellungen stammen aus den Anwendungsfeldern der Angewandten Informatik der Kultur-, Geschichts- und Geowissenschaften. | |
| **Lernziele/Kompetenzen:**<br>Die Studierenden erwerben folgende Kompetenzen:<br>• eine wissenschaftliche Fragestellung in einem vorher festgelegten Themenbereich aufzustellen<br>• diese Fragestellung selbstständig zu bearbeiten und eigene Lösungskonzepte zu entwickeln<br>• eigene Arbeiten zu präsentieren<br>• eine wissenschaftliche Arbeit zu verfassen | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Gruppen- und Einzelbesprechungen: 23 Stunden<br>• Bearbeiten der Praktikumsaufgaben: 57 Stunden<br>• Kolloquiumsvorbereitung: 10 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Allgemeine Informatik-Kenntnisse sowie Interesse an kulturinformatischen Fragestellungen. | **Besondere Bestehensvoraussetzungen:** |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Kulturinformatik**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Christoph Schlieder, Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Inhalte:**<br>Im Rahmen des Bachelor Seminars Kulturinformatik wird ein jeweils von Semester zu Semester wechselndes Themengebiet aus den Kulturinformatik-Modulen Geoinformationssysteme oder Digitale Bibliotheken und Social Computing weiter vertieft. Dies geschieht im Rahmen von Vorträgen und Hausarbeiten zu einer im Vorfeld festgelegten Fragestellung. Dabei steht die selbstständige wissenschaftliche Arbeit im Vordergrund, sowohl schriftlich als auch in der Programmierung. | |

| | |
|---|---|
| **Literatur:**<br>Aktuelle Literatur wird in der Lehrveranstaltung vorgestellt. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 4 Monate<br>Bearbeitungsfrist: 20 Minuten<br>**Beschreibung:**<br>Das Seminarthema wird in Form einer schriftlichen Seminararbeit (Hausarbeit) bearbeitet sowie im Seminar der Arbeitsprozess und das Arbeitsergebnis vorgestellt (Referat). | |

| **Modul KTR-Datkomm-B Datenkommunikation**<br>*Data communication* | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|
| (seit WS17/18)<br>Modulverantwortliche/r: Prof. Dr. Udo Krieger | |
| **Inhalte:**<br>Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Diensteangebots. 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.<br><br>Ferner werden die Grundprinzipien der eingesetzten Übertragungsverfahren, der Übertragungssicherungs- und Steuerungsalgorithmen und der wichtigsten Medienzugriffsverfahren diskutiert, z.B. geläufige Übertragungs- und Multiplextechniken 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.<br><br>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. | |
| **Lernziele/Kompetenzen:**<br>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. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand gliedert sich grob wie folgt:<br>• Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden<br>• Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden<br>• Prüfungsvorbereitung: 35 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>• erfolgreich abgeschlossene Prüfungen der Grundlagenfächer des Bachelorstudiums, insbesondere Einführung in Algorithmen, Programmierung und Software sowie grundlegende Kenntnisse effizienter Algorithmen<br>• gute Programmierkenntnisse in JAVA (oder C++)<br><br>Modul Fortgeschrittene Java Programmierung (DSG-AJP-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| | | |
|---|---|---|
| Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen<br>Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (GdI-MfI-1) - empfohlen | | |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Datenkommunikation**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Udo Krieger<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Inhalte:**<br>Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Diensteangebots. 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 Paketverkehrslenkung dargestellt. Darüber hinaus werden die Grundfunktionen der Transportschicht und ihre exemplarische Umsetzung in TCP erläutert.<br>Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstalltung.<br>**Literatur:**<br>• Lean-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2004<br>• Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 4. Aufl., 2003<br>• Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014<br>• Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2004<br>Weitere Angaben und Erläuterungen erfolgen in der 1. Vorlesung. | **2,00 SWS** |
| **2. Datenkommunkation**<br>**Lehrformen:** Übung | **2,00 SWS** |

**Dozenten:** Prof. Dr. Udo Krieger, Mitarbeiter Informatik, insbesondere Kommunikationsdienste, Telekommunikationssysteme und Rechnernetze
**Sprache:** Deutsch/Englisch
**Angebotshäufigkeit:** WS, jährlich

**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.

Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.

**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, wobei das Bestehen der Modulprüfung die Voraussetzung für die Berücksichtigung dieser individuell erbrachten Bonuspunkte ist. 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 auch ohne diese Zusatzleistung möglich. Das Erreichen der Note 1.0 ist ebenfalls ohne die Erbringung dieser Zusatzleistung möglich.<br><br>Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstalltung.<br><br>Zulässige Hilfsmittel der Prüfung:<br>• Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay | |

| **Modul KTR-MfI-2 Mathematik für Informatik 2 (Lineare Algebra)**<br>*Mathematics for Computer Science 2 (Linear Algebra)* | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|
| (seit WS18/19)<br>Modulverantwortliche/r: Prof. Dr. Udo Krieger | |
| **Inhalte:**<br>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.<br>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:**<br>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. | |
| **Sonstige Informationen:**<br>Das Modul stellt die Grundlagen für Studierende der Angewandten Informatik und Software Systems Science sowie Studierdende im Nebenfach verwandter Bachelorstudiengänge der Fakultät WIAI bereit.<br>Der Arbeitsaufwand gliedert sich grob wie folgt:<br>• Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden<br>• Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden<br>• Prüfungsvorbereitung: 35 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Schulkenntnisse auf dem Niveau eines Mathematik-Vorkurses<br>Modul Mathematik-Vorkurs-Bachelorstudium (KTR-MVK-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>1. - 2. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Mathematik für Informatik 2 (Lineare Algebra)**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Udo Krieger<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **4,00 SWS** |
| **Inhalte:**<br>Die Lehrveranstaltung stellt mathematische Grundlagen der Informatik bereit und ist dem Pflichtbereich der Modulgruppe A1 "Mathematische Grundlage" für Angewandte Informatik/Software Systems Science zugeordnet. Es werden | |

| | |
|---|---|
| grundlegende Methoden und Algorithmen der Gruppen- und Ringtheorie, der linearen Algebra, der Matrizenalgebra, der Lösungstheorie linearer Gleichungssysteme, der Eigenwerttheorie sowie spezifische Anwendungern der Informatik vorgestellt.<br><br>**Literatur:**<br>• A. Steger: Diskrete Strukturen 1, Springer, Heidelberg, 2002.<br>• G. Golub, C.F. van Loan: Matrix Computations, 3ed., Johns Hopkins, 1996.<br>• D. Hachenberger: Mathematik für Informatiker, Pearson, 2008.<br>• J. Liesen, V. Mehrmann: Lineare Algebra, Springer, Bachelorkurs Mathematik, 2. Auflage, 2015.<br>• B. Pareigis: Linear Algebra für Informatiker, Springer, 2000.<br>• M.P.H. Wolff u.a.: Mathematik für Informatik und Bioinformatik, Springer, 2004.<br>• Weitere Literatur wird in der Vorlesung benannt. | |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>Schriftliche Prüfung zu Inhalten der Vorlesung und Übungen im Umfang von 90 Minuten. Zugelassene Hilfsmittel werden in der Lehrveranstaltung bekannt gegeben. | |

| **Modul KTR-Proj Projekt Kommunikationsnetze und -dienste**<br>*Project Communication Networks and Services* | | 6 ECTS / 180 h<br>40 h Präsenzzeit<br>140 h Selbststudium |
|---|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Udo Krieger | | |
| **Inhalte:**<br>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 einzlener Lehrveranstaltungen des Fachgebietes der Professur für Informatik.<br>Die Betriebssystem-Grundausstattung und erforderliche Software-Werkzeuge wie Vyatta-Router, Wireshark, Atheris, RapidStream und andere 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. Details zu den einzelnen Entwicklungsaufgaben, ihren Zielen und Methoden werden in der Vorbesprechung genannt. | | |
| **Lernziele/Kompetenzen:**<br>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.<br>Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten, qualitätsgesicherten Multimediakommunikation und die Fähigkeit, Lösungsvorschläge moderner Dienstarchitekturen im Internet der Zukunft konzipieren, implementieren und sicher beurteilen zu können. | | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Teilnahme an Gruppen- und Einzelbesprechungen: 40 Stunden<br>• Bearbeiten der Projektaufgabe: 120 Stunden<br>• Kolloquiumsvorbereitung: 20 Stunden<br>The module can be selected by Erasmus or exchange students and master students speaking only English. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>• Kenntnisse der Datenkommunikation im Umfang von KTR-Datkomm-B<br>• solide Kenntnisse in JAVA (oder C++)<br>Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **Projekt Kommunikationsnetze und-dienste**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Udo Krieger<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Inhalte:**<br>Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation. Im Mittelpunkt steht die eigenständige, team-orientierte praktische Umsetzung eines Entwicklungsauftrages unter Verwendung des erworbenen Wissens einzlener Lehrveranstaltungen des Fachgebietes der Professur für Informatik.<br>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.<br>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:<br>• Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages),<br>• ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen<br>• und der Darlegung von Zwischenergebnissen bzw.<br>• einem Abschlussbericht mit Abschlusspräsentation<br>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 Vorlesung bereitgestellt.<br>Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstalltung.<br>**Literatur:**<br>Die aktuelle Literatur wird auf der Webseite der Lehrveranstaltung bekanntgegeben. | **4,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Die Bewertung der Prüfungsleistung erfolgt nach Abschluss der Lehrveranstaltung auf folgender Grundlage: | |

| | |
|---|---|
| • Auswertung des von einem Studierenden individuell erstellten schriftlichen Berichts der bearbeiteten Aufgaben im Rahmen einer Einzelarbeit oder der von einem Studierenden im Rahmen einer Gruppenarbeit individuell bearbeiteten Aufgaben, die im schriftlichen Bericht der bearbeiteten Aufgaben unter Verwendung einer eindeutigen Kennzeichung der Urheberschaft dokumentiert werden (mit Bearbeitungsdauer von 4 Monaten)<br>• Vorführung und Erläuterungen der Zusammenhänge einzelner Aufgaben und Ergebnisse im Rahmen einer individuellen Kolloquiumsprüfung im Umfang von 30 Minuten<br><br>Die individuelle Gesamtleistung muss mit der Note "ausreichend" bewertet werden, um die Prüfung zu bestehen.<br><br>Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstalltung. | |

| **Modul KTR-Sem-B Bachelorseminar zu Kommunikationssystemen und Rechnernetzen**<br>*Bachelor Seminar Communication Systems and Computer Networks* | 3 ECTS / 90 h<br>20 h Präsenzzeit<br>70 h Selbststudium |
|---|---|
| (seit WS21/22)<br>Modulverantwortliche/r: Prof. Dr. Udo Krieger | |
| **Inhalte:**<br>Die Studierenden lernen, aktuelle technische Fragestellungen aus den Themenbereichen der Komunikationsnetze und -dienste der neuesten Generation anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen. | |
| **Lernziele/Kompetenzen:**<br>Die Studierenden lernen, aktuelle technische Fragestellungen aus dem Themenfeld der Komunikationsnetze und -dienste anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen. Die Fähigkeit zur kritischen Bewertung komplexer technischer Inhalte nach wissenschaftlichen Grundsätzen der Informatik stelllt ein wichtiges Lernziel dar. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand gliedert sich grob wie folgt:<br>• Präsenzveranstaltungen inkl. Themenvergabe und Besprechungen mit dem Betreuer: 20 Stunden<br>• Bearbeitung des Fachthemas und schriftliche Darstellung: 54 Stunden<br>• Erarbeitung der Präsentation: 16 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Module gemäß der Spezifikationen des Pflichtbereichs sowie solide Kenntnisse der Datenkommunikation<br>Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen<br>Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen | **Besondere Bestehensvoraussetzungen:** |

| **Angebotshäufigkeit:** jährlich nach Bedarf WS und SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** 1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar KTR-Bachelor**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Udo Krieger<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** jährlich nach Bedarf WS und SS<br>**Lernziele:**<br>Die Studierenden lernen, aktuelle Fragestellungen aus dem Themenfeld der Komunikationsnetze und -dienste anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen.<br>**Inhalte:** | **2,00 SWS** |

| | |
|---|---|
| Es werden aktuelle Fragestellungen aus dem Bereich der Kommunikationstechnik und Rechnernetze unter Anleitung bearbeitet . Die aktuelle Themenliste wird auf der Webseite bereitgestellt.<br><br>Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstalltung.<br><br>Die schriftliche Ausarbeitung erfolgt in LATEX, die mündliche Darstellung im Rahmen einer Powerpoint-, LATEX-Beamer oder PDF-Präsentation auf Basis der schriftlichen Ausarbeitung in möglichst freier Rede und logisch korrekter, verständlicher Form.<br><br>**Literatur:**<br>Die aktuelle Literaturliste wird bei der Vorbesprechung bereitgestellt. | |
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Die Gesamtnote ergibt sich zu gleichen Teilen aus der Bewertung der schriftlichen Ausarbeitung (mit Bearbeitungsdauer von maximal 4 Monaten) und des Referats und muss mit mindestens ausreichend bewertet sein.<br><br>Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstalltung. | |

| **Modul KogSys-GAI-B Genderaspekte in der Informatik**<br>*Gender Aspects in Computer Science* | | 3 ECTS / 90 h<br>15 h Präsenzzeit<br>75 h Selbststudium |
|---|---|---|
| (seit WS19/20)<br>Modulverantwortliche/r: Prof. Dr. Ute Schmid | | |
| **Inhalte:**<br>In der Veranstaltung werden theoretische Ansätze und empirische Befunde zu geschlechtsspezifischen Aspekten in der Informatik behandelt, beispielsweise: Geschlechtsstereotype und Studienfachwahl, Informatik in der Schule, Image der Informatik, Einfluss von Rollenmodellen, Barrieren für berufliche Weiterentwicklung. Ein ausgewählter Aspekt kann praktisch bearbeitet werden, beispielsweise: Entwicklung eines Unterrichtsmoduls zur Informatik, Entwickeln einer Image-Kampagne, empirische Erhebung und Analyse. | | |
| **Lernziele/Kompetenzen:**<br>Einblick in Forschungsfragestellungen im Bereich Genderstudies, Verständnis sozialwissenschaftlicher Theorien und empirischer Forschungsmethoden, Kenntnis von Maßnahmen zur Förderung von Mädchen und Frauen in der Informatik, Einblick in Methoden der Evaluationsforschung. | | |
| **Sonstige Informationen:**<br>Das Seminar findet teilweise gemeinsam mit dem Seminar Genderaspekte in der Wirtschaftsinformatik statt, das im Bachelor Wirtschaftsinformatik im Fachgebiet SNA angeboten wird.<br>Der Arbeitsaufwand von 90 Stunden gliedert sich in etwa wie folgt:<br><br>21 Std. Teilnahme an den Präsenzveranstaltungen<br>24 Std. Literaturarbeit, inklusive Vorbereitung von Kurzpräsentationen<br>30 Std. Konzeption und Umsetzung des Praxisteils<br>15 Std. Erstellung der schriftlichen Ausarbeitung | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine Einschränkung | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Genderaspekte in der Informatik**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Ute Schmid<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Inhalte:**<br>siehe Modulbeschreibung | |
| **Literatur:** | |

| | |
|---|---|
| wird in der Veranstaltung bekannt gegeben | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Beschreibung:**<br>Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema. | |

| **Modul KogSys-KI-B Einführung in die Künstliche Intelligenz**<br>*Introduction to Artificial Intelligence* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Ute Schmid | |

**Inhalte:**

Das Modul bietet einen Einstieg in die grundlegenden Konzepte und Methoden der Künstlichen Intelligenz. Zentrale Themen sind Suchen und Problemlösen, Spiele und Constraints, Wissensrepräsentation und Logik, Schlussfolgern und Planen. Ausgewählte Aspekte weiterführender Themen aus den Bereichen Unsicheres Wissen, Maschinelles Lernen, Sprache und Kommunikation, Bildanalyse und Robotik werden behandelt. Neben der Vermittlung von theoretischen Grundlagen wird die Umsetzung von KI-Algorithmen in Scheme und Prolog vermittelt. In der Vorlesung werden auch Geschichte der KI, interdisziplinäre Bezüge, insbesondere zu Philosophie und Psychologie sowie ethische Fragen der KI angesprochen.

Liste der Themen:
• Problemlösen und Suche
• Suchalgorithmen für Spiele
• Ansätze der Wissensrepräsentation
• Aussagen- und Prädikatenlogik
• Inferenz in Logik erster Stufe
• Nicht-klassische Logiken
• Planung
• Maschinelles Lernen
• Sprachverarbeitung
• Objekt- und Szenenerkennung

**Lernziele/Kompetenzen:**

- Grundlegende Konzepte und Problemstellungen der KI definieren und erklären können
- Einfache KI-Algorithmen auf konkrete – auch neue – Problemstellungen anwenden können
- Problemstellungen formal, insbesondere mit Mitteln der Logik modellieren können
- Grundzüge von KI-Programmiertechniken (insbesondere funktionale und logische Programmierung) beherrschen

**Sonstige Informationen:**

Die Folien (Vorlesung und Übung) sowie die Prüfungsunterlagen sind in englischer Sprache verfügbar. Weitere Materialien sind überwiegend in englischer Sprache.

Der Zeitaufwand gliedert sich in 45h Präsenzzeit und 135h Selbststudium.

Zeitaufwand aufgeschlüsselt:

• 22,5h Vorlesung + 30h Nachbereitung

• 22,5h Übung + 75h Bearbeitung von Übungsaufgaben

• 30h Klausurvorbereitung

**Zulassungsvoraussetzung für die Belegung des Moduls:**

(außer für FÜM Kognitive Künstliche Intelligenz)

| | | |
|---|---|---|
| • GdI-MfI-1 (Mathematik für Informatik 1)<br>• DSG-EiAPS-B (Einführung in Algorithmen, Programmierung und Softwaretechnik) | | |
| **Empfohlene Vorkenntnisse:**<br>Kenntnisse in den folgenden Bereichen, zugehörige Module in Klammern:<br>• Algorithmen und Datenstrukturen (AI-AuD-B)<br>• Introduction to Functional Programming (GdI-IFP-B oder GdI-IFP-M)<br>• Grundlagen der Theoretischen informatic (GdI-GTI-B)<br>• Lineare Algebra (xAI-MML-M, KTR-MfI-2-B) | | **Besondere Bestehensvoraussetzungen:**<br>Keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Einführung in Künstliche Intelligenz**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Ute Schmid<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>Siehe Modulbeschreibung<br>**Inhalte:**<br>Präsentation und Diskussion der Inhalte (siehe Modulbeschreibung), insbesondere theoretische und konzeptionelle Aspekte.<br>**Literatur:**<br>Stuart Russel und Peter Norvig (2021, 4. Auflage). Artificial Intelligence, A Modern Approach (AIMA). Prentice Hall. | **2,00 SWS** |
| **Prüfung**<br>schriftliche Modulprüfung (Klausur) / Prüfungsdauer: 105 Minuten<br>**Beschreibung:**<br>Die Prüfungsdauer beinhaltet eine Lesezeit von 15 Minuten, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können. In der Klausur können 90 Punkte erzielt werden. Die Klausur ist bestanden, wenn mindestens 40 Prozent erreicht werden.<br>Im Semester werden freiwillige Studienleistungen (Übungsblätter) ausgegeben. Durch die freiwillige Bearbeitung der Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus den optionalen Studienleistungen bestanden ist.<br>Zu Beginn der Lehrveranstaltung wird bekannt gegeben:<br>• Art und Anzahl der Studienleistungen<br>• Umfang (Anzahl an erreichbaren Punkten) der Studienleistungen<br>• Bearbeitungsdauer der Studienleistungen<br>Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden. Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialen, Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay. | |

| | |
|---|---|
| Die Aufgabenstellungen in der Klausur sind auf Deutsch und Englisch verfasst. | |

| **Lehrveranstaltungen** | |
|---|---|
| **Einführung in Künstliche Intelligenz**<br>**Lehrformen:** Übung<br>**Dozenten:** Bettina Finzel<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>Siehe Modulbeschreibung<br>**Inhalte:**<br>Praktische Vertiefungen zu den Inhalten der Vorlesung:<br>• Wiederholen und Vertiefen von theoretischen Konzepten, die in der Vorlesung vorgestellt wurden<br>• Simulation von Algorithmen der Suche, der logischen Inferenz, der Planung und des maschinellen Lernens (händisch und programmatisch)<br>• Aufgaben zur Wissensmodellierung und zur Modellierung logischer Welten<br>• Berechnen von Heuristiken<br>• Wahrscheinlichkeitsrechnung<br>• Erarbeiten von Beispielanwendungen in denen Künstliche Intelligenz zum Einsatz kommen kann<br>• Präsentation und Diskussion von Aufgabenlösungen | **2,00 SWS** |

| **Modul KogSys-ML-B Einführung in Maschinelles Lernen**<br>*Introduction to Machine Learning* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Ute Schmid | |

**Inhalte:**

Das Modul bietet einen Einstieg in die grundlegenden Konzepte und Methoden des maschinellen Lernens. Dabei wird ein breiter Überblick über symbolischer, neuronale und statistische Ansätze, deren mathematische Grundlagen, sowie algorithmische Umsetzung gegeben. In der Vorlesung werden auch Geschichte des maschinellen Lernens, interdisziplinäre Bezüge, insbesondere zu Philosophie und Psychologie sowie ethische Fragen des maschinellen Lernens angesprochen.

Liste der Themen:
• Grundkonzepte: Inductive Biases, Hypothesenraum
• Evaluation von gelernten Modellen
• Entscheidungsbäume und Random Forests
• Induktive Logische Programmierung
• Künstliche Neuronale Netze
• Support Vector Machines, Kernels, Margins
• Ausgewählte Ansätze des Deep Learning: Embeddings, Convolutional Neural Networks, Transformer, VAEs
• Instanzbasierte Methoden
• Bayes'sches Lernen
• Lernen von Sequenzen
• Reinforcement Learning
• Weitere Aspekte: Erklärbarkeit, Neuro-symbolische Ansätze, Knowledge-informed Machine Learning, Human-in-the-loop Learning.

**Lernziele/Kompetenzen:**
• Grundlegende Konzepte und zentrale Ansätze des maschinellen Lernens erläutern und anwenden können
• Zentrale symbolische, neuronale und statistische Algorithmen des Klassifikationslernens auf gegebene Daten anwenden können
• Die Eignung gegebener Daten für Algorithmen des Klassifikationslernen beurteilen können
• Die Güte gelernter Modelle beurteilen können
• Gemeinsamkeiten und Unterschiede zwischen menschlichem und maschinellem Lernen erörtern können
• Moderne Bibliotheken für maschinelles Lernen in relevanten Programmiersprachen, insbesondere Python, verwenden können

**Sonstige Informationen:**
Die Folien (Vorlesung und Übung) sowie die Prüfungsunterlagen sind in englischer Sprache verfügbar. Weitere Materialien sind überwiegend in englischer Sprache.

Der Zeitaufwand gliedert sich in 45h Präsenzzeit und 135h Selbststudium.

Zeitaufwand aufgeschlüsselt:

• 22,5h Vorlesung + 30h Nachbereitung

• 22,5h Übung + 75h Bearbeitung von Übungsaufgaben

• 30h Klausurvorbereitung

**Zulassungsvoraussetzung für die Belegung des Moduls:**
• DSG-EiAPS-B (Einführung in Algorithmen, Programmierung und Softwaretechnik)
• GdI-MfI-1 (Mathematik für Informatik 1)

| **Empfohlene Vorkenntnisse:** | **Besondere Bestehensvoraussetzungen:** |
|---|---|
| Kenntnisse in den folgenden Bereichen, zugehörige Module in Klammern:<br>• Objektorientierte Programmierung (DSG-JaP-B, KInf-IPKult-E)<br>• Lineare Algebra (xAI-MML-M, xAI-MML-B, KTR-MfI-2-B) | keine |

| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** 1 Semester |
|---|---|---|

**Lehrveranstaltungen**

**1. Lernende Systeme (Machine Learning)** — **2,00 SWS**

**Lehrformen:** Vorlesung
**Dozenten:** Prof. Dr. Ute Schmid
**Sprache:** Deutsch/Englisch
**Angebotshäufigkeit:** WS, jährlich

**Lernziele:**
Siehe Modulbeschreibung

**Inhalte:**
Präsentation und Diskussion der Inhalte (siehe Modulbeschreibung), insbesondere theoretische und konzeptionelle Aspekte.

**Literatur:**
Mitchell, Machine Learning, McGraw-Hill, 1997.
Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, 2012.
Goodfellow et al., Deep Learning, MIT Press, 2016.
Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

**2. Lernende Systeme (Machine Learning)** — **2,00 SWS**

**Lehrformen:** Übung
**Dozenten:** Johannes Langer
**Sprache:** Deutsch/Englisch
**Angebotshäufigkeit:** WS, jährlich

**Lernziele:**
Siehe Modulbeschreibung

**Inhalte:**
Praktische Vertiefungen zu den Inhalten der Vorlesung:

| | |
|---|---|
| • Wiederholen und Vertiefen von theoretischen Konzepten, die in der Vorlesung vorgestellt wurden<br>• Implementation von in der Vorlesung vorgestellten Algorithmen<br>• Handsimulation von in der Vorlesung vorgestellten Algorithmen<br>• Berechnung von für maschinelles Lernen relevanten Metriken, zur Evaluation oder als Teile von Algorithmen<br>• Erarbeiten von Beispielanwendungen in denen maschinelles Lernen zum Einsatz kommen kann<br>• Präsentation und Diskussion von Aufgabenlösungen<br><br>**Literatur:**<br>siehe Vorlesung | |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten<br>**Beschreibung:**<br>Die Prüfungsdauer **beinhaltet eine Lesezeit von 15 Minuten**, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.<br><br>In der Klausur können 90 Punkte erzielt werden. Die Klausur ist bestanden, wenn mindestens 40 Prozent erreicht werden.<br><br>Im Semester werden freiwillige Studienleistungen (Übungsblätter) ausgegeben. Durch die freiwillige Bearbeitung der Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus den optionalen Studienleistungen bestanden ist. Zu Beginn der Lehrveranstaltung wird bekannt gegeben:<br>• Art und Anzahl der Studienleistungen<br>• Umfang (Anzahl an erreichbaren Punkte) der Studienleistungen<br>• Bearbeitungsdauer der Studienleistungen<br><br>Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.<br><br>Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialen, Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay.<br><br>Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben. | |

| **Modul KogSys-Sem-B Bachelorseminar Kognitive Systeme**<br>*Bachelor Seminar Cognitive Systems* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Ute Schmid | | |
| **Inhalte:**<br>Aufbauend auf den in den Vorlesungen und Übungen des Faches Kognitive Systeme erworbenen Kenntnissen und Fertigkeiten wird im Seminar die eigenständige Erarbeitung und Präsentation eines Themengebiets auf der Basis von wissenschaftlicher Literatur eingeübt. Die Seminarthemen sind aus dem Bereich Künstliche Intelligenz, beispielsweise:<br>• Explainable AI<br>• Human-in-the-Loop Learning<br>• AI and Education<br>• Neuro-symbolic AI<br>• Representation Learning<br>• Ultra Strong Machine Learning<br>• Generative AI | | |
| **Lernziele/Kompetenzen:**<br>• Einarbeitung in eine spezielle Fragestellung aus dem Bereich Künstliche Intelligenz anhand wissenschaftlicher Literatur mit Fokus auf einem konkreten Algorithmus oder einer konkreten Methode anhand eines vorgegebenen Textes<br>• Suche nach wissenschaftlicher Literatur und Bewertung von Qualität und Relevanz<br>• Mündliche Präsentation einer wissenschaftlichen Arbeit<br>• Abfassen eines Forschungspapiers nach vorgegebenem Format entlang einer Forschungsfrage in Englisch<br>• Diskussion von wissenschaftlichen Arbeiten im Seminar | | |
| **Sonstige Informationen:**<br>Zeitaufwand:<br>• 22,5h Präsenz<br>• 2,5h persönliche Besprechungstermine mit dem Dozenten/der Dozentin<br>• 30h Erarbeitung der Literatur<br>• 10h Vorbereitung der Präsentation<br>• 25h Abfassen der schriftlichen Ausarbeitung | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Mindestens eines der folgenden:<br>• KogSys-KI-B (Einführung in die Künstliche Intelligenz)<br>• KogSys-ML-B (Einführung in Maschinelles Lernen) | | |
| **Empfohlene Vorkenntnisse:**<br>keine<br>Modul Einführung in die Künstliche Intelligenz (AI-KI-B) - empfohlen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| Lehrveranstaltungen | |
|---|---|
| **Bachelorseminar Kognitive Systeme**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Ute Schmid, Johannes Langer, Bettina Finzel<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Lernziele:**<br>Siehe Modulbeschreibung<br>**Inhalte:**<br>Siehe Modulbeschreibung<br>**Literatur:**<br>wird zu Beginn des Seminars bekanntgegeben | **2,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Schriftliche Ausarbeitung zu dem im Seminar bearbeiteten Thema.<br>Die Prüfungssprache wird in der ersten Lehrveranstaltung bekannt gegeben. | |

| **Modul MI-Sem-B Bachelorseminar zur Medieninformatik**<br>*Media Informatics Seminar [Bachelor]* | 3 ECTS / 90 h |
|---|---|
| (seit SS20)<br>Modulverantwortliche/r: Prof. Dr. Andreas Henrich | |
| **Inhalte:**<br>Aufbauend auf den in den Vorlesungen und Übungen des Faches Medieninformatik erworbenen Kenntnissen und Fertigkeiten wird in diesem Seminar die eigenständige Erarbeitung und Präsentation von Themengebieten auf Basis der Literatur verfolgt. | |
| **Lernziele/Kompetenzen:**<br>Im Seminar werden die Fähigkeiten im Bereich der kritischen und systematischen Literaturrecherche und -betrachtung ebenso weiterentwickelt wie die Kompetenzen in der Präsentation von Fachthemen (schriftlich und im Vortrag) sowie deren Diskussion. | |
| **Sonstige Informationen:**<br>Die Lehrveranstaltung wird in **Deutsch** durchgeführt. Zahlreiche Quellen und Dokumentationen sind aber auf **Englisch** verfasst. Vorträge und Ausarbeitungen können in Deutsch oder Englisch verfasst werden. Teilnehmerinnen und Teilnehmer müssen an Fachdiskussionen auf Deutsch und Englisch teilnehmen können.<br><br>Der **Arbeitsaufwand** für dieses Modul gliedert sich typischerweise in folgende Bereiche:<br>• Teilnahme an den Präsenzveranstaltungen (Themenvergabe, Besprechungen, Präsentationen): ca. 20 Stunden<br>• Literaturrecherche ...: ca. 25 Stunden<br>• Vorbereitung der Präsentation: ca. 15 Stunden<br>• Erstellen der schriftlichen Ausarbeitung: ca. 30 Stunden | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Kenntnisse entsprechend den unten angegebenen Modulen. Details werden in jedem Semester in der Vorbesprechung oder der Vorankündigung bekannt gegeben.<br><br>Modul Bachelor AI Teil-Modulgruppe Wissenschaftliches Arbeiten (AI-WissArb-B) - empfohlen<br>Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen<br>Modul Web-Technologien (MI-WebT-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Medieninformatik**<br>**Lehrformen:** Proseminar<br>**Dozenten:** Prof. Dr. Andreas Henrich, Mitarbeiter Medieninformatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |

| | |
|---|---|
| **Inhalte:**<br>Im Seminar werden wechselnde aktuelle Forschungsthemen zu den Inhalten der Lehrveranstaltungen bearbeitet. Dabei sind im Regelfall Aspekte mehrerer Lehrveranstaltungen relevant.<br>**Literatur:**<br>wird jeweils zu Beginn der Veranstaltung bekannt gegeben | |
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>**Beschreibung:**<br>Hausarbeit und Referat zu dem im Seminar vom Teilnehmer bzw. von der Teilnehmerin bearbeiteten Thema, inkl. Diskussion | |

| | |
|---|---|
| **Modul MI-WAIAI-B Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik**<br>*Introduction to Academic Research & Writing for Computer Science and Applied Computer Science* | 3 ECTS / 90 h<br>22 h Präsenzzeit<br>68 h Selbststudium |

(seit WS24/25)
Modulverantwortliche/r: Prof. Dr. Andreas Henrich

**Inhalte:**
Im Modul werden wesentliche Methoden und Grundsätze des wissenschaftlichen Arbeitens in Informatik und Angewandter Informatik behandelt:

- Wissenschaft, Ethik, Forschung
- Wissenschaftliche Arbeiten
- Wissenschaftliches Arbeiten & Schreiben
- Projektmanagement
- Vortragen & Präsentieren
- Evaluation & empirische Methoden

**Lernziele/Kompetenzen:**
Die Studierenden erwerben die Fähigkeit, selbstständig den Kenntnisstand in einem wissenschaftlichen Teilgebiet der Informatik oder Angewandten Informatik an Hand von Literaturempfehlungen zu überprüfen, zu erweitern und sich mit dem Stand der Forschung vertraut zu machen.

Sie erwerben Kenntnisse zu fachspezifischen Methoden der Literatursuche und lernen Systeme kennen, die bei der Erstellung wisenschaftlicher Arbeiten unterstützen.

Allgemein werden die Grundlagen zum wissenschaftlichen Arbeiten, zum Verfassen wissenschaftlicher Arbeiten und zum Präsentieren vermittelt und eingeübt.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| | | |
|---|---|---|
| **Empfohlene Vorkenntnisse:**<br>Grundlegende Kenntnisse der Informatik und Angewandten Informatik.<br>Das Modul sollte ab dem 3. Semester besucht werden und vor der ersten Belegung eines Seminars. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 3. | **Minimale Dauer des Moduls:**<br>1 Semester |

| | |
|---|---|
| **Lehrveranstaltungen** | |
| **Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Andreas Henrich, Mitarbeiter Medieninformatik<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Lernziele:**<br>siehe Modulbeschreibung<br>**Inhalte:** | **2,00 SWS** |

| | |
|---|---|
| siehe Modulbeschreibung<br><br>**Literatur:**<br>• Helmut Balzert, Marion Schröder, Christian Schäfer: *Wissenschaftliches Arbeiten*, W3L GmbH, 2. Auflage, Taschenbuch, 2011, ISBN-13: 978-3868340341<br>• Manuel René Theisen, *Wissenschaftliches Arbeiten: Erfolgreich bei Bachelor- und Masterarbeit*, Vahlen, Auflage: 17 (2017), ISBN-13: 978-3800653829 | |

| | |
|---|---|
| **Prüfung**<br>Portfolio / Bearbeitungsfrist: 4 Monate<br>**Beschreibung:**<br>**Gegenstand** des Portfolios sind alle Inhalte von Vorlesung und Übung.<br><br>Im Semester werden studienbegleitend (in der Regel zweiwöchentlich) Aufgabenstellungen ausgegeben und besprochen, deren individuelle Bearbeitung im Portfolio zu dokumentieren ist. | |

| **Modul MI-WebT-B Web-Technologien**<br>*Web Technologies* | 6 ECTS / 180 h<br>45 h Präsenzzeit<br>135 h Selbststudium |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Andreas Henrich | |
| **Inhalte:**<br>Nach einer 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:**<br>Studierende sollen methodische, konzeptuelle und praktische Fähigkeiten und Fertigkeiten zur Erstellung von Web-Applikationen erwerben. Besonderes Augenmerk wird dabei auf aktuelle Web Technologien gelegt. Die Studierenden sind nach Abschluss des Moduls in der Lage, Web-Anwendungen selbständig mit gängigen Frameworks und Techniken zu entwickeln und hierzu passende Technologie Stacks auszusuchen. | |
| **Sonstige Informationen:**<br>Die Lehrveranstaltungen werden in **Deutsch** durchgeführt. Zahlreiche Quellen und Dokumentationen der Systeme sind aber auf **Englisch**.<br><br>Der **Arbeitsaufwand** für dieses Modul gliedert sich grob wie folgt:<br>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)<br>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden<br>• Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb)<br>• Bearbeiten der 3 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb)<br>• Prüfungsvorbereitung und Prüfung: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff) | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Grundkenntnisse der Informatik und zu Medienformaten, wie sie z. B. in den unten angegebenen Modulen erworben werden können. Insbesondere sind auch Kenntnisse in einer imperativen oder objektorientierten Programmiersprache erforderlich.<br><br>Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen<br>Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>3. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **1. Web-Technologien** | **2,00 SWS** |

**Lehrformen:** Vorlesung
**Dozenten:** Prof. Dr. Andreas Henrich
**Sprache:** Deutsch
**Angebotshäufigkeit:** SS, jährlich

**Inhalte:**
Die Veranstaltung betrachtet die Aufgabenfelder, Konzepte und Technologien zur Entwicklung von Web-Anwendungen. Folgende Bereiche bilden dabei die Schwerpunkte der Veranstaltung:

- Das Web: Einführung, Architektur, Protokolle ...
- Sprachen zur Beschreibung von Webseiten: HTML & CSS
- Client-Side Scripting: Basics, AJAX, Bibliotheken
- Server-Side Scripting: Node.js, PHP und weiterführende Konzepte
- Frameworks auf Client- und Serverseite
- Sicherheit von Web-Anwendungen
- CMS, LMS, SEO & Co.

**Literatur:**
aktuelle Literatur wird in der Veranstaltung bekannt gegeben

**2. Web-Technologien** — **2,00 SWS**
**Lehrformen:** Übung
**Dozenten:** Mitarbeiter Medieninformatik
**Sprache:** Deutsch
**Angebotshäufigkeit:** SS, jährlich

**Inhalte:**
praktische Aufgaben zum Stoff der Vorlesung

**Literatur:**
siehe Vorlesung

**Prüfung**
schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten
**Beschreibung:**
**Gegenstand** der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten).

In der **Klausur** können 90 Punkte erzielt werden.

In der Prüfungsdauer von 105 Minuten ist eine **Lesezeit** von 15 Minuten enthalten, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.

Im Semester werden studienbegleitend 3 **Teilleistungen** (schriftliche Hausarbeiten) in der Übung ausgegeben und besprochen, deren Abgabe freiwillig ist. Für jede Teilleistung stehen in der Regel 4 Wochen als Bearbeitungszeit zur Verfügung. Die abgegebenen 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. | |

| **Modul MII-SemHRI-B Bachelorseminar Mensch-Roboter-Interaktion**<br>*Bachelor Seminar Human-Robot Interaction* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS23/24)<br>Modulverantwortliche/r: Prof. Dr. Markus Rickert | | |
| **Inhalte:**<br>Eine große Herausforderung bei der Interaktion zwischen Mensch und Roboter liegt in der gemeinsamen Bearbeitung von komplexen Aufgaben. Ein Robotersystem muss in der Lage sein, die aktuelle Umgebung wahrzunehmen, Intentionen und Ziele vorauszusehen, mittels verschiedener Modalitäten mit seinem Gegenüber zu kommunizieren und physisch mit der Umgebung und dem Menschen zu interagieren. Das Seminar gibt einen allgemeinen Überblick über die Herausforderungen bei dieser Form der Interaktion zwischen Mensch und Roboter. | | |
| **Lernziele/Kompetenzen:**<br>Studierende lernen, sich ein ausgewähltes Thema anhand aktueller Fachliteratur zu erarbeiten und mündlich wie auch schriftlich zu präsentieren. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Keine Vorkenntnisse erforderlich; Kenntnisse in Robotik, kognitiven Systemen, Bild- oder Sprachverarbeitung können von Vorteil sein. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Mensch-Roboter-Interaktion**<br>**Lehrformen:** Seminar<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |

| **Prüfung** | |
|---|---|
| Referat mit schriftl. Hausarbeit / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 4 Monate | |

| **Modul MOBI-DBS-B Datenbanksysteme**<br>*Database Systems* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Daniela Nicklas | | |
| **Inhalte:**<br>Das Modul vermittelt eine systematische Einführung in das Gebiet der Datenbanksysteme. | | |
| **Lernziele/Kompetenzen:**<br>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 durch das Entity Relationship Model. Die Studierenden erlernen die Grundlagen der Sprache SQL und können mit SQL Datenbankschemata generieren sowie zugehörige Datenbanken aufbauen und manipulieren. Sie verstehen die Grundlagen von Transaktionssystemen. Schließlich sammeln sie erste Erfahrungen im Umgang mit realen Datenbankverwaltungssystemen. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Datenbanksysteme**<br>**Lehrformen:** Vorlesung, Übung<br>**Dozenten:** Prof. Dr. Daniela Nicklas<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS | **4,00 SWS** |
| **Lernziele:**<br>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 durch das Entity Relationship Model. Die Studierenden erlernen die Grundlagen der Sprache SQL und können mit SQL Datenbankschemata generieren sowie zugehörige Datenbanken aufbauen und manipulieren. Sie verstehen die Grundlagen von Transaktionssystemen. Schließlich sammeln sie erste Erfahrungen im Umgang mit realen Datenbankverwaltungssystemen. | |
| **Inhalte:**<br>• Datenbank-Konzepte und -Architektur<br>• Modellierung von Datenbanken: Das ER- und EER-Modell<br>• Das relationale Modell<br>• Relationale Algebra<br>• SQL (DDL und DML) | |

| | |
|---|---|
| • Normalisierung und Normalformen<br>• Datenbanken im Mehrbenutzerbetrieb: Transaktionssysteme und Recovery<br>• Alternative Entwicklungen im Bereich Datenbanken<br><br>**Literatur:**<br>Date C.J.: An Introduction to database systems. 8th Edition, Addison-Wesley, Reading, Massachusetts 2003<br><br>Elmasri & Navathe: Grundlagen von Datenbanksystemen, Pearson, 2002 | |
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten<br>**Beschreibung:**<br>Zentrale Klausur. Gegenstand der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich Teilleistungen; siehe unten).<br><br>Die Prüfung besteht aus 7 Aufgaben, von denen die besten 6 gewertet werden. Die Prüfungsdauer beinhaltet eine Lesezeit von 15 Minuten, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.<br><br>Im Semester werden studienbegleitend Teilleistungen ausgegeben und besprochen, deren Abgabe freiwillig ist. Für die Bearbeitung dieser Teilleistungen können Bonuspunkte vergeben werden. Die Anzahl und Bedinungen der zu erreichenden Bonuspunkte sowie deren Umrechnungsfaktor in mögliche Klausurpunkte werden in der ersten Übungsstunde bekannt gegeben.<br><br>Ist die Klausur bestanden (in der Regel sind hierzu 50% der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten, ggf. umgerechneten, Punkte zusätzlich angerechnet. Die Note 1,0 ist auch ohne Punkte aus Teilleistungen erreichbar. | |

| **Modul MOBI-Proj-B Bachelor Project Mobile Software Systems**<br>*Bachelor Project Mobile Software Systems* | 6 ECTS / 180 h |
|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Daniela Nicklas | |
| **Inhalte:**<br>Applications of mobile software systems, which are taken from current research activities in mobile, context-aware systems and data stream management, are carried out in part individually and in part in small teams of students, from conception, via theoretical and/or practical realization, to evaluation. In particular, the project concerns the development of sound concepts pertaining to the task to be addressed under the given project constraints. This requires studying the current research literature and relevant approaches on the project's topic.<br>An example of a project task would be the conceptual development, the prototypic implementation, and the case-study-driven evaluation of a small sensor-based, mobile system, which would require knowledge from the modul MOBI-DSC-M Data streams and event processing.<br>The tasks in the project will be tailored to Bachelor level. | |
| **Lernziele/Kompetenzen:**<br>Students will gain knowledge and practical experience with the problems that arise when carrying out practical research and software develeopment in the field of mobile computing and sensor-based applications. This covers modeling methods, project mangagement, and data management developement. | |
| **Sonstige Informationen:**<br>Dauer der Lehranstaltung 1 Semester | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Programming skills (Java preferred), e.g. from the module "DSG-AJP-B"; Software project management, e.g. from the module "SWT-SWL-B Software Engineering Lab"; Scientific research and writing, e.g. from the module "MI-WAIAI-B Wissenschaftliches Arbeiten" or SSS-SRW-M Scientific Research on Writing for Master´s Students; Relational databases and SQL, e.g. from the module MOBI-DBS-B (Datenbanksysteme) | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS<br>**Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelor project Mobile Software Systems**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Daniela Nicklas<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS<br>**Inhalte:** | **4,00 SWS** |

| | |
|---|---|
| Durchführung des Projekts.<br><br>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.<br><br>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. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Kolloquium<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen.<br>Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden zu Beginn einer jeden Lehrveranstaltung von der Projektleiterin bzw. dem Projektleiter bekannt gegeben.<br>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). The term of the project report and of the colloquium will be announced at the beginning of each course by the project leader. | |

| **Modul MOBI-SEM-B Bachelor-Seminar Mobile Software Systems**<br>*Bachelor-Seminar Mobile Software Systems* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS19/20)<br>Modulverantwortliche/r: Prof. Dr. Daniela Nicklas | | |
| **Inhalte:**<br>Guided study and presentation of a topic from Mobile Software Systems / Mobility. The overarching theme of the seminar and example topics will be presented at the beginning of the seminar. The participants will either choose an example topic or define an individual topic within that theme and will contribute to the seminar with presentations and discussions. | | |
| **Lernziele/Kompetenzen:**<br>Scientific writing and presentation; insight in selected topics; the ability to structure and present a scientific topic. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Scientific research and writing, e.g. from the module "IAIWAI-B Wissenschaftliches Arbeiten" or "SSS-SRW-M Scientific Research on Writing for Master´s Students". | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Mobile Software Systems**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Daniela Nicklas<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Inhalte:** | **2,00 SWS** |
| **Prüfung**<br>Hausarbeit mit Referat<br>**Beschreibung:**<br>Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen. Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Referates werden zu Beginn einer jeden Lehrveranstaltung von der Projektleiterin bzw. dem Projektleiter bekannt gegeben.<br>Production of a written report on the seminar topic (Assignment/Hausarbeit). Discussion of this seminar report in the context of the wider seminar topic (presentation/Referat). The term of the seminar report and of the presentation will be announced at the beginning of each course by the project leader. | |

| **Modul PSI-DatSchu-B Datenschutz**<br>*Data Protection* | | 3 ECTS / 90 h |
|---|---|---|
| (seit SS23)<br>Modulverantwortliche/r: N.N.<br>Weitere Verantwortliche: Prof. Dr. Dominik Herrmann | | |
| **Inhalte:**<br>Grundbegriffe und Grundlagen des Datenschutzrechts, Datenschutz im öffentlichen/nicht-öffentlichen Bereich, betrieblicher Datenschutz, Grundprinzipien, insbesondere Datenminimierung, Pseudonymisierung, Löschung, Rechte der Betroffenen, Verantwortliche, Auftragsverarbeiter, Privacy by design and default, Datensicherheit, Grundlagen der Informationssicherheit, Datenpannen, Beschäftigtendatenschutz, Spannungsverhältnis Informationssicherheit und Datenschutz, Drittlandtransfer, Aktuelle Fallgestaltungen | | |
| **Lernziele/Kompetenzen:**<br>Grundkenntnisse des Datenschutzrechts (DSGVO, BDSG, TTDSG, ePrivacyVO) im IT-Bereich, die Fähigkeit zur eigenständigen Lösung einfacher Fälle und Kenntnisse der besonderen Fragen der Anwendung des Datenschutzrechts auf Fallgestaltungen der elektronischen Datenverarbeitung (Cloud Computing, Social Media, internationaler Datentransfer). Weiterhin sind die Studierenden in der Lage, Grundkenntnisse der Schnittstellen zur Informations- und IT-Sicherheit (Art. 32 DSGVO) zu verstehen und anzuwenden. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine; alle notwendigen Inhalte werden in der Veranstaltung erarbeitet. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Datenschutz**<br>**Lehrformen:** Vorlesung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |
| **Lernziele:**<br>siehe Modulbeschreibung | |
| **Literatur:**<br>Zwingend nötig ist es mit einer aktuellen Gesetzessammlung (Beck: Datenschutzrecht – DatSchR) zu arbeiten, um die Vorschriften mitlesen zu können. Lehrbuch: Marie-Theres Tinnefeld/Benedikt Buchner/Thomas Petri/ Hans-Joachim Hof, Einführung in das Datenschutzrecht. Datenschutz und Informationsfreiheit in europäischer Sicht, Berlin (De Gruyter), aktuelle Auflage | |

| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten | |
|---|---|

| **Modul PSI-EDS-B Ethics for the Digital Society**<br>*Ethics for the Digital Society* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Dominik Herrmann | | |
| **Inhalte:**<br>This module introduces students to fundamental concepts of ethics and their application to techniques that shape the digital society. It discusses the influence of current and upcoming technologies and their implications from an ethical perspective. The lecture is accompanied by a series of case studies, which focus on a concrete problem that is to be analyzed by the participants. Topics include decision making in autonomous systems and systems that employ so-called artificial intelligence, the reliability and dependability of computer systems, and privacy aspects of information systems. | | |
| **Lernziele/Kompetenzen:**<br>Participants will be able to reflect on their actions as a scientist as well as a computer professional. They learn how to evaluate the trade-offs that are inherent in new technologies and how to design information systems in ways that support the needs of a digital society. Successful participants will obtain the ability to apply ethical thinking to novel problems and potential solutions. | | |
| **Sonstige Informationen:**<br>The module is taught in English unless all participants are fluent in German. There may be a small number of guest lectures that is taught in German.<br>During the semester multiple case studies will be published. Participants will be asked to submit essays or solutions (small programs) discussing ethical aspects of those case studies. Essays will be peer-reviewed by other participants. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Ethics for the Digital Society**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Dominik Herrmann<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, jährlich | **2,00 SWS** |
| **Lernziele:**<br>cf. module description | |
| **Inhalte:**<br>cf. module description | |
| **Literatur:**<br>• Ibo van de Poel and Lamber Royakkers: Ethics, Technology, and Engineering – an Introduction | |

| | |
|---|---|
| • Jay Quinn: Ethics for the Information Age<br>• Herman T. Tavani: Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing | |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 80 Minuten<br>**Beschreibung:**<br>The exam time includes a reading time of 20 minutes.<br><br>The exam questions will be in English. The questions can be answered in English or German. The content that is relevant for the exam consists of the content presented in the lecture and in the case studies.<br><br>The maximum number of points that can be achieved in the exam is 100. Participants can collect up to 10 bonus points during the semester. Details regarding the number of assignments, the number of points per assignment, and the type of assignments will be announced in the first lecture.<br><br>If the points achieved in the exam are sufficient to pass the exam on its own (generally, this is the case when at least 50 points have been obtained), the bonus points will be added to the points achieved in the exam. The grade 1.0 can be achieved without the bonus points. | |

| **Modul PSI-EiRBS-B Einführung in Rechner- und Betriebssysteme**<br>*Introduction to Computer Architecture and Operating Systems* | 6 ECTS / 180 h |
|---|---|
| (seit SS24 bis WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Michael Engel | |

**Inhalte:**

Das 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 sowie die wesentlichen Komponenten von Rechner- und Betriebssystemen. Behandelt werden insbesondere der Aufbau und die Funktionsweise eines minimalen Rechners (von-Neumann-Architektur) sowie die Darstellung von Daten (Zahlen und Zeichenketten) im Rechner sowie ihre Speicherung und Verarbeitung. Darüber hinaus werden die wesentlichen Komponenten der Systemsoftware (insbes. Prozess-Scheduling und Speicherverwaltung) erläutert und deren Zusammenspiel mit der Rechnerarchitektur aufgezeigt. Die Themen werden anhand von Modellen, marktgängigen Programmiersprachen (insbes. C) und aktuellen Rechner- und Betriebssystemen (RISC V und Linux) behandelt. Abschließend wird ein erster Einblick in Rechnernetze und Aspekte der Systemsicherheit gegeben.

Im Rahmen der Veranstaltung erarbeiten sich die Studierenden praktische Kenntnisse im Umgang mit der Linux-Kommandozeile sowie der Assembly-Programmierung. Diese Inhalte erarbeiten sich die Studierenden anhand von bereitgestellten Materialien (Skript und weitere Literatur) und Aufgaben primär im Selbststudium.

**Lernziele/Kompetenzen:**

Studierende haben einen ersten Überblick über die Gebiete der Informatik und kennen die grundlegenden Begriffe und Methoden der Informatik sowie die wichtigsten in der Informatik verwendeten Techniken. 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. Die Studierenden sind dazu in der Lage, auf der Linux-Kommandozeile grundlegende Datenverarbeitungsaufgaben zu erledigen sowie einfache Algorithmen mit Assembly-Instruktionen zu kodieren.

**Sonstige Informationen:**

Der Arbeitsaufwand von 180 Stunden verteilt sich ausgehend von einem 14 Arbeitswochen dauernden Semester in etwa wie folgt:

- 21 Std. Vorlesungsteilnahme
- 21 Std. Übungsteilnahme
- 56 Std. Bearbeiten von wöchentlichen Übungsaufgaben, d.h. ca. 4 Std./Woche
- 42 Std. Vor- und Nachbereitung von Vorlesung und Übung (Videos, Skript, Recherchen), d.h. ca. 3 Std./Woche)
- 40 Std. Vorbereitung auf und Zeit für die Abschlussklausur

Bei diesem Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

keine

| | | |
|---|---|---|
| **Empfohlene Vorkenntnisse:**<br>Es sind keine Vorkenntnisse erforderlich. Erste Erfahrungen im Umgang mit der Linux-Kommandozeile sowie einer Programmiersprache (etwa C oder Java) sind hilfreich. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Einführung in Rechner- und Betriebssysteme**<br>**Lehrformen:** Vorlesung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>vgl. Modulbeschreibung<br>**Literatur:**<br>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 diese Bücher erfolgreich absolviert werden. Darüber hinaus wird ein ausführliches Skript zur Verfügung gestellt.<br>• Tanenbaum, A.S./Austin, T.: Structured Computer Organization. Addison-Wesley, 2012 (6th).<br>• Arpaci-Dusseau , R.H./Arpaci-Dusseau, A.C.: Operating Systems: Three Easy Pieces, 2018.<br>• Silberschatz, A./Gagne, G./Galvin, P B.: Operating Systems Concepts. John Wiley and Sons, 2012 (9th) | **2,00 SWS** |
| **2. Einführung in Rechner- und Betriebssysteme (Übung)**<br>**Lehrformen:** Übung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>In der Übung werden die wichtigsten Konzepte der Vorlesung an theoretischen und praktischen Beispielen (z.B. anhand eines Linux-Systems) veranschaulicht und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema vertieft. | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 110 Minuten<br>**Beschreibung:**<br>Gegenstand der Klausur sind die Inhalte der Vorlesung, des Skripts und der Übungen. Die Prüfungsdauer beinhaltet eine Lesezeit von 20 Minuten. | |

| **Modul PSI-IntroSP-B Introduction to Security and Privacy**<br>*Introduction to Security and Privacy* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Dominik Herrmann | |
| **Inhalte:**<br>This module introduces students to fundamental concepts in the fields of information security and the protection of privacy. It provides a broad overview over the most relevant topics from a technical perspective. The focus lies on practical issues that have to be considered when professional and personal information systems are built and operated. | |
| **Lernziele/Kompetenzen:**<br>Successful students will know the mathematical background behind basic cryptographic primitives and be able to explain fundamental concepts of information security and privacy, including classical attacks and defenses. They will be able to apply their knowledge when implementing simple attack programs as well as building and operating defensive techniques. | |
| **Sonstige Informationen:**<br>This module is taught in English. It consists of a lecture and tutorials. During the course of the tutorials there will be theoretical and practical assignments (task sheets). Assignments and exam questions can be answered in English or German.<br><br>Workload breakdown:<br>• Lecture: 22.5 hours (2 hours per week)<br>• Tutorials: 22.5 hours (2 hours per week)<br>• Preparation and studying during the semester: 30 hours<br>• Assignments: 67.5 hours<br>• Preparation for the exam (including the exam itself): 37.5 hours | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>It is strongly recommended to take this module only after successful completion of introductory courses on computer science on programming, algorithms, data structures, computer architecture, and operating systems.<br><br>Prospective PSI-IntroSP-B participants should be familiar with fundamentals of computer architecture (binary representation of strings and numbers in computers, bitwise operators (such as XOR), operation of a CPU, basics of assembly language), operating systems (memory layout and process management), and computer networks (basic IP routing and addressing, TCP/IP connection establishment). Also, basic familiarity with the Linux command line is recommended.<br><br>Moreover, basic familiarity with common web technologies (HTTP, HTML, JavaScript) as well as relational database systems and SQL is a recommended prerequisite. | **Besondere Bestehensvoraussetzungen:**<br>keine |

| | | |
|---|---|---|
| Finally, participants should have working knowledge in at least one programming language (e.g., Python, C, or Java) so that they can write small tools for automation purposes on demand.<br>Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen<br>Modul Grundlagen der Rechnerarchitektur und Betriebssysteme (Inf-GRABS-B) - empfohlen | | |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:** 1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Introduction to Security and Privacy**<br>**Lehrformen:** Vorlesung<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Lernziele:**<br>cf. module description<br>**Inhalte:**<br>Selected topics<br>• Security Terminology (protection goals, attacker and attack types)<br>• Authentication and Authorization Fundamentals<br>• Software Security in C and Assembly (e.g., buffer overflows, selected defenses)<br>• Cryptography (e.g., historic ciphers, symmetric and asymmetric cryptosystems, Diffie-Hellman key exchange, TLS protocol)<br>• Network Security (spoofing, denial of service, authentication protocols, intrusion detection systems)<br>• Web Security (attacks and defenses related to the OWASP Top 10 including SQL injections and Cross Site Scripting)<br>• Privacy and Techniques for Data Protection (re-identification risks, anonymization networks, k-anonymity, the idea of differential privacy)<br>**Literatur:**<br>Selected books:<br>• A. Shostack: Threat Modelling<br>• W. Stallings: Computer Security: Principles and Practice<br>• J. Erickson: Hacking: The Art of Exploitation | **2,00 SWS** |
| **2. Introduction to Security and Privacy**<br>**Lehrformen:** Übung<br>**Sprache:** Englisch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Inhalte:**<br>In the tutorials, participants work on tasks and assignments to obtain practical skills related to the information security and privacy topics covered in the lecture. | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Testat / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>In the intermediate examination (e-exam), participants demonstrate that they master the practical skills acquired by completing the assignments. | |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 120 Minuten<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>To be admitted to the final examination (e-exam), participants must have passed the intermediate exam (e-exam).<br>**Beschreibung:**<br>The exam time includes a reading time of 30 minutes.<br>Details about the requirements for admission to the written examination will be announced in the first lecture.<br>The content that is relevant for the exam consists of the content presented in the lecture and tutorials. The exam questions are in English. The exam questions can be answered in English or German. | |

| **Modul PSI-ProjectPAD Project Practical Attacks and Defenses**<br>*Project Practical Attacks and Defenses* | 6 ECTS / 180 h |
|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Dominik Herrmann | |

**Inhalte:**

Breaking into information systems is exciting, but impractical due to ethical and legal concerns. However, offensive competences and adversarial thinking are essential to build secure systems. In this project students will get the opportunity to acquire practical security skills in a dedicated training environment.

The goal of this project is to build and extend the "Insekta" platform. This web-based tool provides a frontend for virtual machines that can be used to study selected topics in security and privacy on one's own and at one's own pace.

This project is offered together with PSI-ProjectCAD-M, which focuses on conceptually more complex attacks and defenses.

The participants of the project familiarize themselves with security weaknesses in information systems and apply this knowledge to develop vulnerable services which others can use for training. To this end, participants form groups, read about attacks and defenses in textbooks and research papers, and discuss various options to implement them. Instructors will provide extensive and on-demand support to enable the participants to implement a vulnerable service that can be exploited to learn about a particular vulnerability.

Besides implementing vulnerable services, the participants prepare training materials, which consist of questions and tasks to test one's knowledge as well as step-by-step instructions. These training materials may also contain interactive elements for an improved learning experience.

The project also takes into account attacks on privacy, e.g., re-identifying individuals in anonymized datasets and communication networks, tracking users on the Internet, inferring sensitive attributes from seemingly harmless data traces, as well as mitigations, e.g., depersonalization strategies and differential privacy mechanisms. Here, practical activities consist in the preparation of datasets and scripts for analysis.

**Lernziele/Kompetenzen:**

Successful students will be able to describe attacks and defenses from textbooks and research papers in easily understandable form. They will also be able to carry out selected attacks in practice and implement defenses with a programming language of their choice.

**Sonstige Informationen:**

This project is taught in English, unless all participants are fluent in German. The workload of this project is equivalent to 180 hours.

Workload breakdown:

- 10 hrs: Getting familiar with the platform
- 30 hrs: Reading papers and researching security vulnerabilities
- 15 hrs: Preparing the talk (including time for attendance of other talks)
- 70 hrs: Implementing the vulnerable service and defenses
- 55 hrs: Writing training material and documentation

Note that there is another project (PSI-ProjectCAD-M) with a workload equivalent to 270 hours.

| | | |
|---|---|---|
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Students in bachelor and master programs can participate in this project.<br><br>Participants should be familiar with basic concepts in information security and privacy, which can be acquired, for instance, by taking the module "Introduction to Security and Privacy" (PSI-IntroSP-B). This includes basic knowledge about the commonly used security terminology, common types of malware and attacks, buffer overflows and related attacks, cryptography, network security, web security, and concepts of privacy.<br><br>Moreover, participants should have practical experience with at least one scripting or programming language such as Python or Java. Experience with Linux environments, web technologies, and network protocols is recommended. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| | |
|---|---|
| **Lehrveranstaltungen** | |
| **Project Practical Attacks and Defenses**<br>**Lehrformen:** Projektseminar<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS | **4,00 SWS** |
| **Lernziele:**<br>cf. module description | |
| **Inhalte:**<br>Potential topics include:<br>• web security (injection flaws and other issues mentioned in the OWASP Top 10)<br>• network security (such as DNS cache poisoning and rebinding attacks)<br>• security issues in C programs (buffer overflows, etc.)<br>• cryptography (low-level attacks on ciphers, high-level attacks on protocols, e.g., TLS)<br>• business logic failures<br>• misconfigurations<br>• attacks on availability (denial of service)<br>• attacks on privacy (such as inference, tracking, re-identification, fingerprinting)<br>• privacy defenses (such as k-anonymity, related concepts, differential privacy) | |
| **Literatur:**<br>Literature will be announced at the beginning of the project. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regular attendance at project meetings.<br>**Beschreibung:**<br>The module examination consists of two parts: Firstly, the participants submit a written report (in English) that includes the source code of the vulnerable service and the training material. Secondly, the participants give a talk in which they defend their work (in English; in German if all participants are fluent in German) by presenting theoretical and practical aspects of their vulnerable service as well as relevant mitigations. The maximum number of points that can be achieved in the module examination is 100.<br><br>Optionally, participants can submit intermediary results (in English) to collect up to 20 bonus points. If the module examination is passed on its own (generally, this is the case when at least 50 points are obtained), the bonus points will be added to the points achieved in the module examination. The grade 1.0 can be achieved without the bonus points. Details regarding the number of optional submissions during the semester, their type, the points per submission, and the respective deadlines will be announced in the first session of the project. | |

| **Modul PSI-Sem-B Seminar Security and Privacy Foundations**<br>*Seminar Security and Privacy Foundations* | 3 ECTS / 90 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Dominik Herrmann | |
| **Inhalte:**<br>In this seminar, participants learn how to work with scientific literature (books, papers, and articles), how to write in the language used in science, and how to give a talk about a topic covering security and/or privacy foundations. They give a talk summarizing their findings and write them up in a term paper.<br>While participants are expected to perform the actual research on their own, the instructors provide extensive support throughout the seminar. The following questions will be addressed: how to approach a topic, how to find relevant literature, how to read a paper efficiently, how to write a seminar report, and how to give a good talk?<br>The actual topics are subject to change. A list of available topics is made available in the first week of the teaching period via the seminar's VC course. | |
| **Lernziele/Kompetenzen:**<br>Participants learn to find, read, and summarize scientific texts. They also learn to assess statements and to discuss them critically. Finally, they learn to write scientific texts and to present their results in a talk. | |
| **Sonstige Informationen:**<br>The default language in this seminar is English, unless all participants are fluent in German. | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Participants are encouraged to obtain basic knowledge in software engineering before taking this module. Basic knowledge in information security and privacy is recommended but not strictly required.<br>Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Seminar Security and Privacy Foundations**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Dominik Herrmann<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS | **2,00 SWS** |
| **Lernziele:**<br>cf. module description | |
| **Inhalte:**<br>cf. module description | |
| **Literatur:**<br>Selected text books: | |

| | |
|---|---|
| • Alley: The Craft of Scientific Writing<br>• Anderson: Security Engineering<br>• Pfleeger et al.: Security in Computing<br>• Stallings & Brown: Computer Security: Principles and Practice<br>• Strunk & White: The Elements of Style<br><br>Other relevant literature is presented in the first session. | |

| | |
|---|---|
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 2 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Continuous attendance in the seminar sessions is mandatory, cf. §9 (10) APO.<br>**Beschreibung:**<br>The module examination consists of two parts, a term paper (in English) and a talk (in English; in German if all participants are fluent in German). The maximum number of points that can be achieved in the module examination is 100. Details regarding the number of points that can be achieved in the talk and in the report will be announced in the first session of the project.<br><br>Optionally, participants can participate in writing and presentation labs, where they can submit intermediary results (in English) or give mock presentations (in English). Participants can thereby earn 20 bonus points. If the module examination is passed on its own (generally, this is the case when at least 50 points are obtained), the bonus points will be added to the points achieved in the module examination. The grade 1.0 can be achieved without the bonus points. | |

| **Modul SWT-FPS-B Foundations of Program Semantics**<br>*Foundations of Program Semantics* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen | | |
| **Inhalte:**<br>This theory module offers students a mathematical perspective on sequential, imperative computer programs and concurrent, distributed software. It studies the formal semantics of programs which enables their analysis and verification. | | |
| **Lernziele/Kompetenzen:**<br>On completion of this module, students will be able to understand and apply core foundational concepts of, and techniques behind, program semantics and verification, in the context of both sequential and concurrent systems. | | |
| **Sonstige Informationen:**<br>The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German.<br>The total workload of 180 hrs. is split approximately as follows:<br>• 60 hrs. attending lectures (Vorlesungen)<br>• 30 hrs. attending practicals (Übungen)<br>• 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources<br>• 30 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium) | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in discrete mathematics, logics and algebra. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Foundations of Program Semantics**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Gerald Lüttgen<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, jährlich | **4,00 SWS** |
| **Inhalte:**<br>Students will be introduced to the foundations of program semantics and their applications to program verification. Particular emphasis will be put on mathematical theories for reasoning about sequential and concurrent systems.<br>The following topics will be covered:<br>Part I: Mathematical Foundations<br>• Inductive definitions and proofs<br>• Orders, functions and fixed points | |

- Algebraic structures, equivalences and congruences
- Algebraic laws and logic systems

Part II: Sequential, Imperative Programs

- The imperative language IMP
- Natural, structural operational and denotational semantics
- The Hoare calculus

Part III: Concurrent, Distributed Software

- The process calculus CCS
- Strong and weak bisimulation
- Algebraic laws and axiomatization

**Literatur:**

- Bruni, R. and Montanari, U., Models of Computation. Springer, 2017.
- Milner, R. Communication and Concurrency. Prentice Hall, 1989.
- Nielson, H. R. and Nielson, F., Semantics with Applications: An Appetizer. Springer, 2007.
- Loeckx, J. and Sieber, K. The Foundations of Program Verification, 2nd ed. Wiley, 1987.
- Steffen, B., Rüthing, O. and Huth, M. Mathematical Foundations of Advanced Informatics. Springer, 2018.
- Davey, B. A. and Priestley, H. A. Introduction to Lattices and Order, 2nd ed. Cambridge University Press, 2002.

**2. Foundations of Program Semantics** — **2,00 SWS**

**Lehrformen:** Übung
**Dozenten:** Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
**Sprache:** Englisch/Deutsch
**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**
The practicals (Übungen) cover pen-and-paper exercises that will deepen the concepts and techniques taught in the lectures (Vorlesungen). Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students.

**Literatur:**
- see the corresponding lectures -

**Prüfung**
Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten
Bearbeitungsfrist: 3 Wochen
**Beschreibung:**
Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen). The assignment is set in English; students may answer in either English or German.

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). The examination language is either English or German and may be chosen by the student at the colloquium. | |

| **Modul SWT-FSE-B Foundations of Software Engineering**<br>*Foundations of Software Engineering* | 6 ECTS / 180 h |
|---|---|
| (seit SS23)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen | |
| **Inhalte:**<br>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:**<br>Students will receive an introduction to the common problems and paradigms in, and foundations of, software development. They will also gather conceptional 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. | |
| **Sonstige Informationen:**<br>The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German.<br>The total workload of 180 hrs. is split approximately as follows:<br>• 45 hrs. attending lectures (Vorlesungen)<br>• 30 hrs. reviewing the lectures, including researching and studying material from additional sources<br>• 45 hrs. attending practicals (Übungen)<br>• 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources<br>• 30 hrs. preparing for the written exam (Klausur) | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in Computer Science, as well as knowledge in programming in Java and in algorithms and data structures. | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** SS, jährlich<br>**Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Foundations of Software Engineering**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Gerald Lüttgen<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **3,00 SWS** |
| **Inhalte:**<br>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, model-driven and pattern-based development, and software testing. Students are also introduced to specific aspects of agile software development.

**Literatur:**

- Sommerville, I. Software Engineering, 10th ed. Pearson, 2016.
- Robertson, S. and Robertson, J. Mastering the Requirements Process, 3rd ed. Addison-Wesley, 2012.
- Cohn, M. User Stories Applied. Addison-Wesley, 2004.
- Stevens, P. and Pooley, R. Using UML - Software Engineering with Objects and Components, 2nd ed. Addison-Wesley, 2006.
- Freeman, E., Robson, E., Sierra, K. and Bates, B. Head First Design Patterns, 2nd ed. O'Reilly, 2020.
- Gamma, E., Helm, R., Johnson, R. and Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Design. Prentice Hall, 1994.

Further literature will be announced in the lectures.

**2. Foundations of Software Engineering** — **3,00 SWS**

**Lehrformen:** Übung
**Dozenten:** Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
**Sprache:** Englisch/Deutsch
**Angebotshäufigkeit:** SS, jährlich

**Inhalte:**
The practicals (Übungen) exercise and deepen the conceptual knowledge transferred via the lectures (Vorlesungen), and relay practical knowledge in software engineering.

**Literatur:**
- see the corresponding lectures -

**Prüfung**
schriftliche Prüfung (Klausur) / Prüfungsdauer: 120 Minuten
**Beschreibung:**
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 set in English, while answers may be provided in either English or German. The exam is passed if at least 50% of the available points are reached.

| **Modul SWT-SEM-B Seminar Software Engineering and Programming Languages (Bachelor)**<br>*Seminar in Software Engineering and Programming Languages (Bachelor)* | 3 ECTS / 90 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen | |
| **Inhalte:**<br>Current topics in software engineering and programming languages. | |
| **Lernziele/Kompetenzen:**<br>Students will compile and acquire current topics in software engineering and programming languages by carrying out and documenting a guided literature survey, and by preparing and delivering a coherent, comprehensible presentation to their peers. | |
| **Sonstige Informationen:**<br>The main language of instruction is English. The seminar may be delivered in German if all participating students are fluent in German. Regular participation in the presentations is required.<br>The total workload of 90 hrs. is split approximately as follows:<br>• 5 hrs. introductory tutorials on the seminar's topic and on scientific methods<br>• 20 hrs. consultations and presentations (Referate), including discussions<br>• 25 hrs. literature research and familiarization and evaluation of literature<br>• 40 hrs. working on the assignment (schriftliche Hausarbeit) and preparation for the presentation (Referat) | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in software engineering and programming languages. | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Software Engineering and Programming Languages (Bachelor)**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>Various current topics in software engineering and programming languages, which complement and/or extend the technical and methodological aspects of the degree programme's modules related to these fields.<br>**Literatur:**<br>Literature will be allocated according to the topics to be discussed. | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 40 Minuten<br>Bearbeitungsfrist: 8 Wochen<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regular participation in the seminar.<br>**Beschreibung:**<br>Presentation (Referat) on the topic assigned to the student, including a discussion.<br><br>Assignment (schriftliche Hausarbeit) consisting of a written report on the topic assigned to the student. | |

| **Modul SWT-SWL-B Software Engineering Lab**<br>*Software Engineering Lab* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen | | |
| **Inhalte:**<br>Small teams of students will conduct a software project, starting from a brief problem description.<br>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:**<br>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. | | |
| **Sonstige Informationen:**<br>The main language of instruction is English. The practicals may be delivered in German if all participating students are fluent in German. A regular attendance of team meetings and active participation is required throughout.<br>The total workload of 180 hrs. is split approximately as follows:<br>• 20 hrs. attending meetings of the student's team with the lecturer (Dozent) on planning, coordination and feedback<br>• 10 hrs. attending the accompanying practicals/tutorials (Übungen/Tutorials) on software tools<br>• 130 hrs. conducting the team project<br>• 20 hrs. working on the written assignment (schriftliche Hausarbeit) and preparing for the colloquium (Kolloquium) | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in Computer Science and Software Engineering, as well as knowledge in Java programming and in programming in the small. | | **Besondere Bestehensvoraussetzungen:** |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Software Engineering Lab**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, jährlich | **4,00 SWS** |
| **Inhalte:** | |

Each team will carry out a software project, 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:**

- Tudose, C., Tahchiev, P., Leme, F., Massol, V. and Gregory, G. JUnit in Action, 3rd ed. Manning Publications, 2020.
- 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, Pearson, 2001.
- Cohn, M. User Stories Applied. Addison-Wesley, 2004.

See the description of the module "Foundations of Software Engineering (SWT-FSE-B)" for further literature.

**Prüfung**

Hausarbeit mit Kolloquium, schriftliche Hausarbeit mit Kolloquium /
Prüfungsdauer: 45 Minuten
Bearbeitungsfrist: 2 Wochen

**Zulassungsvoraussetzung zur Modulprüfung:**

Regular participation in the associated practicals, including the participation in programming tasks.

**Beschreibung:**

Written assignment (schriftliche Hausarbeit) involving the compilation of a written project report in English or German language by each team, which shall cover the following topics:

- A description of the team’s produced artefacts, plus 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.

The submission deadline and the details of the required content and format of this report will be announced at the beginning of the semester.

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. The colloquium can be held electively in English or German language.

Because this module involves a team effort, the examination can only be resit in a winter semester.

| **Modul SWT-SWP-B Software Engineering Project**<br>*Software Engineering Project* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen | | |
| **Inhalte:**<br>Students run their own "company" to develop new and to maintain and extend existing software products for customers. Thereby, students deepen their understanding of agile development practices in self-organizing teams and become familiar with modern software engineering practices and state-of-the-art software technologies such as frameworks for backend architectures, web-based frontends, and user authentication. | | |
| **Lernziele/Kompetenzen:**<br>By the end of this module, students will be able to<br>• Work within self-organizing, agile software development teams,<br>• Comprehend source code and documentation written by others,<br>• Apply advanced software architectures, frameworks, and patterns,<br>• Use state-of-the-art tools for robust software development, and<br>• Understand the value of software quality assurance for long-living software. | | |
| **Sonstige Informationen:**<br>Depending on the participants involved, the main language of instruction is German or English. Regular attendance of team meetings and active participation is required throughout.<br>The total workload of 180 hrs. is split approximately as follows:<br>• 10 hrs. attending tutorials<br>• 20 hrs. attending meetings with customers and lecturers on planning, coordination and feedback<br>• 30 hrs. attending meetings within student teams<br>• 100 hrs. working on multiple software products<br>• 20 hrs. writing the product reports and journal (schriftliche Hausarbeit) and preparing for the colloquium (Kolloquium) | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>None | | |
| **Empfohlene Vorkenntnisse:**<br>Practical software engineering skills such as acquired in a first software engineering team project. Advanced knowledge in one or more programming languages. | | **Besondere Bestehensvoraussetzungen:**<br>None |
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 5. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Software Engineering Project (Übung/Practical)**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** SS, jährlich | **4,00 SWS** |

| | |
|---|---|
| **Lernziele:**<br>See "learning outcomes / competences" above.<br><br>**Inhalte:**<br>The first three weeks of the module are used for tutorials on the technologies applied in the module. Afterwards, five two-week sprints are conducted, in which students have regular meetings in order to plan sprints, monitor progress, review results, and discuss software engineering processes and techniques. Students present the results of each sprint in a sprint review meeting, which is attended by customers and lecturers.<br><br>**Literatur:**<br>Literature will be announced at the beginning of the semester and depend on the technologies to be used. | |
| **Prüfung**<br>Hausarbeit mit Kolloquium, Software Engineering Project (schriftliche Hausarbeit mit Kolloquium) / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 12 Wochen<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regular participation in the associated practicals, including the participation in programming tasks.<br>**Beschreibung:**<br>The "schriftliche Hausarbeit" comprises a written report in German or English language on each software product, to which all students involved in the product's development shall contribute, plus one individual report per student in form of a journal that summarizes the student's activities and the lessons learned throughout the semester. Details on deadlines, required content, and format of the "schriftliche Hausarbeit" will be announced at the beginning of the semester.<br>The "Kolloquium" discusses the "schriftliche Hausarbeit" as well as the artifacts created throughout the semester. It may take place individually or, with an according adjustment of the examination's duration, in groups of students, and can be held electively in German or English language. | |

| **Modul SYSNAP-Project-B Projekt Systemnahe Programmierung**<br>*Project Systems Programming* | 6 ECTS / 180 h |
|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Michael Engel | |

**Inhalte:**
Students work (in groups) on a small yet realistic project to develop a standalone piece of system software that is not solvable in acceptable time by a single student. Hence, besides

- basic literature research to find approaches to solve the problem(s) at hand and to get used to the state-of-the-art technology required,
- analyzing, designing, architecting, programming and testing the practical solution,

skills such as planning, delegating and organizing work in groups are practiced.
Note: The topics of this master project are - compared to bachelor projects - more advanced and lead to advanced skills in the development of operating systems, machine-level and assembler programming as well as debugging.

**Lernziele/Kompetenzen:**
Students learn how to

- work independently and in groups on selected problems using the knowledge and skills provided by other modules,
- work with state-of-the-art tools and refer to recent scientific literature to look for problem solutions,
- architect and implement an operating system kernel interacting with emulators and real hardware,
- read, understand and apply data sheets as well as processor and peripheral user manuals
- document and present their work in an understandable manner to others,
- interact with others to discuss pros and cons of different solution approaches,
- organize work in groups, esp., how to delegate work, to fix interfaces and work under time constraints.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:**<br>Module Inf-GRABS-B | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Projekt Systemnahe Programmierung**<br>**Lehrformen:** Projektseminar<br>**Dozenten:** Prof. Dr. Michael Engel<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, SS | **4,00 SWS** |
| **Lernziele:**<br>see module description | |
| **Inhalte:** | |

| | |
|---|---|
| see module description<br><br>**Literatur:**<br>Based on the concrete project topics literature will be provided at the start of the semester. | |
| **Prüfung**<br>Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 3 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>As this is a project in groups and the topic of the examination is the project work of each student, each student has to declare which part of the project and report is due to his own work.<br>**Beschreibung:**<br>Project report and developed software based on the project work indicating which are the on achievements during the project.<br><br>Oral examination concerning the technologies used in the project as well as the work of the group a student belongs to with an emphasis on her or his own work. | |

| **Modul SYSNAP-SEM-B Seminar System Software**<br>*Seminar System Software* | | 3 ECTS / 90 h |
|---|---|---|
| (seit SS24)<br>Modulverantwortliche/r: Prof. Dr. Michael Engel | | |
| **Inhalte:**<br>Current topics in system software, including operating systems, hypervisors, just-in-time compilation and hardware-software interfacing. Topics cover the full spectrum of research topics in these fields, from the analysis, design, implementation and evaluation of current system software, to the discussion and evaluation of novel research proposals.<br>Bachelor students will also receive material about working with scientific literature, the use of scientific methods, as well as preparing the seminar report and presentation. | | |
| **Lernziele/Kompetenzen:**<br>Students will compile and acquire current topics in operating systems by independently carrying out and documenting a literature survey, and by preparing and delivering a coherent, comprehensible presentation to their peers. Students will also be able to scientifically discuss topics in system software with their peers. | | |
| **Sonstige Informationen:**<br>Participation in the LaTeX course organized by the Fachschaft WIAI is recommended. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Basic knowledge in system software, machine-level programming and computer architecture and in the subject matter of the seminar. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Seminar**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Michael Engel<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, SS<br>**Lernziele:**<br>cf. module description<br>**Inhalte:**<br>cf. module description<br>**Literatur:**<br>Recent papers on system software related to the respective focus of the seminar, announced at the start of the semester. | **2,00 SWS** |
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regular participation in the group meetings | |

| | |
|---|---|
| **Beschreibung:**<br>Review of a written elaboration on the most important aspects of the topic, including a correct list of references.<br>Participation in peer reviewing the other participants;<br>free holding of a a presentation based on presentation documents including discussion of the contents with the seminar participants. | |

| **Modul SYSNAP-SNAP-B Systemnahe Programmierung**<br>*Systems Programming* | 6 ECTS / 180 h |
|---|---|
| (seit WS25/26)<br>Modulverantwortliche/r: Prof. Dr. Michael Engel | |

**Inhalte:**

***Das Modul wird voraussichtlich erst ab dem WS 2026/27 angeboten!***

Das Modul bietet einen vertieften Einblick in die Strukturen und Algorithmen sowie den Code auf Systemebene mit dem Ziel, das Verständnis systemnaher Software durch eine vertikale Analyse der Ebenen Hardware, Systemsoftware und Compiler zu vertiefen. Am Beispiel realer Systemsoftware und moderner Rechnerarchitekturen wird die Interaktion von Hard- und Software, die durch Systemsoftware koordiniert und durch Compiler und Laufzeitumgebungen realisiert wird, betrachtet und die Abbildung typischer System- und Programmierabstraktionen auf die Funktionalität von Hardwarekomponenten anhand realer Codebeispiele analysiert.

Schwerpunkte bilden hierbei die Realisierung typischer moderner Algorithmen aus verschiedensten Aufgabenbereichen von Betriebssystemen, Compilern und Laufzeitumgebungen, wie z.B. Synchronisation, Scheduling, Speicherverwaltung, Übersetzung von Hochsprachenkonstrukten in Assembler sowie nicht-funktionale Eigenschaften von Systemen wie Zuverlässigkeit oder Echtzeitfähigkeiten. Hierbei werden verschiedenste Anwendungsbereiche von eingebetteten Systemen über Desktop- und Serversysteme bis hin zum Hochleistungsrechnen analysiert. Als Beispiele kommen typische Komponenten des verwendeten xv6-Betriebssystems wie auch realer Code aus Open Source-Systemen wie L4, Linux, BSD oder macOS/Darwin auf einer modernen RISC-Prozessorplattform wie RISC-V oder ARM v8 zum Einsatz.

Ergänzend wird der für systemnahe Programmierung relevante Teil des Softwareentwicklungsprozesses betrachtet. Hier erfolgt ein Einblick in den Übersetzungsvorgang systemnaher Sprachen (z.B. C oder Rust), die Aufgaben von Linker und Loader, Binärformate und Handhabung von Bibliotheken und schließlich ein Überblick über Debugging-Methoden, die für Software auf Systemebene relevant sind.

Im Rahmen der Veranstaltung erarbeiten sich die Studierenden praktische Kenntnisse im Verstehen und eigenen Entwickeln von Code auf Systemebene und der Interaktion des Codes mit Hardware in Emulatoren sowie realen Systemen.

**Lernziele/Kompetenzen:**

Aufbauend auf Grundkenntnissen aus dem Modul EiRBS lernen Studierende hier die innere Funktionsweise von Systemsoftware und Compilern anhand von Analysen, Experimenten sowie der selbstständigen Entwicklung von Systemsoftwarekomponenten kennen. Studierende sollen ein Verständnis für systemnahe Abstraktionen und verschiedene existierende Konzepte für deren Realisierung entwickeln sowie Abstraktionen im Rahmen eines vorgegebenen Systems verstehen, analysieren, debuggen sowie eigenständig entwickeln können.

Im Rahmen der Übungen entwickeln die Studierenden durch Lesen, Analyse und Benchmarking nicht-funktionaler Eigenschaften, Debuggen und eigenen Implementationen von Funktionalität des xv6-Kernels praktische Fähigkeiten, die die Basis für eine Forschungs- oder Entwicklungstätigkeit im Bereich der Systemsoftware darstellen.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

Bestandene Prüfung in Inf-GRABS-B oder PSI-EiRBS-B

| | | |
|---|---|---|
| **Empfohlene Vorkenntnisse:**<br>Grundkenntnisse in Programmierung, Rechnerarchitektur und Betriebssysteme, z.B. aus Inf-GRABS-B oder PSI-EiRBS-B und EIAPS. | | **Besondere Bestehensvoraussetzungen:**<br>– tbd – |
| **Angebotshäufigkeit:** WS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 5. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Systemnahe Programmierung**<br>**Lehrformen:** Vorlesung und Übung<br>**Dozenten:** Prof. Dr. Michael Engel<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, jährlich<br>**Lernziele:**<br>- tbd -<br>**Inhalte:**<br>*- tbd -*<br>***Das Modul wird voraussichtlich erst ab dem WS 2026/27 angeboten!***<br>**Literatur:**<br>- tbd - | **4,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>Portfolio / Prüfungsdauer: 3 Monate<br>Bearbeitungsfrist: 3 Monate<br>**Beschreibung:**<br>Das Portfolio besteht aus der Bearbeitung mehrerer theoretischer Aufgaben sowie praktischer Programmieraufgaben im Bereich der systemnahen Programmierung. | |

| **Modul SoSySc-Thesis-B Bachelorarbeit in Software Systems Science**<br>*Bachelor Thesis in Software Systems Science* | | 12 ECTS / 360 h |
|---|---|---|
| (seit SS19 bis SS24)<br>Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen<br>Weitere Verantwortliche: Professorinnen und Professoren der Informatik | | |
| **Inhalte:**<br>Das Modul Bachelorarbeit hat einen Umfang von 12 ECTS-Punkten und beinhaltet eine schriftliche Prüfung in Form der Bachelorarbeit. Das Thema der Bachelorarbeit ist einem der in Anhang 2 der Prüfungsordnung genannten Fächer zu entnehmen.<br>Auf Antrag der Prüfungskandidatin bzw. des Prüfungskandidaten kann vom Prüfungsausschuss auch ein Thema aus einem anderen Fach zugelassen werden. In diesem Fall ist nachzuweisen, dass das gestellte Thema inhaltlich der Software Systems Science entnommen ist. | | |
| **Lernziele/Kompetenzen:**<br>Mit der Bachelorarbeit soll der Nachweis erbracht werden, dass die Prüfungskandidatin bzw. der Prüfungskandidat in der Lage ist, das gestellte Thema selbstständig mit wissenschaftlichen Methoden zu bearbeiten, indem sie erlerntes Fachwissen unter Verwendung wissenschaftlicher Methoden auf eine vorgegebene Forschungsfrage anwenden.<br>Die Prüfungskandidatin bzw. der Prüfungskandidat lernt, sich weitgehend selbstständig in eine wissenschaftliche Fragestellung einzuarbeiten. Sie erarbeiten eigeninitiativ eine wissenschaftliche Arbeit und wenden das im Studium erworbene Wissen gezielt und reflektiert an. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>Die Zulassung setzt voraus, dass Module im Umfang von mindestens 120 ECTS-Punkten erfolgreich absolviert wurden. | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>ab dem 6. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Prüfung**<br>schriftliche Hausarbeit / Bearbeitungsfrist: 4 Monate | |
|---|---|

| **Modul Stat-B-01 Methoden der Statistik I**<br>*Statistical Methods I* | 6 ECTS / 180 h |
|---|---|

(seit SS22)
Modulverantwortliche/r: Prof. Dr. Timo Schmid

**Inhalte:**
Die Veranstaltung „Methoden der Statistik I“ beschäftigt sich im ersten Teil mit der deskriptiven Statistik von ein- und zweidimensionalen empirischen Verteilungen. Ein Schwerpunkt liegt auf Verfahren, mit deren Hilfe Daten übersichtlich dargestellt und aufbereitet werden. Daneben vermittelt die Veranstaltung Kenntnisse zu aussagekräftigen Maßzahlen zur Charakterisierung von Daten, insbesondere Lageparameter, Streuungsmaße und Korrelationskoeffizienten.

Im zweiten Teil der Veranstaltung stehen Grundlagen der Wahrscheinlichkeitsrechnung im Vordergrund. Im Mittelpunkt steht dabei die Beschreibung zufälliger Vorgänge mithilfe von parametrischen Zufallsvariablen. Aufbauend auf dem Konzept von Wahrscheinlichkeiten wird der Begriff der Zufallsvariablen hergeleitet. Neben der Behandlung grundlegender Konzepte und Definitionen werden wichtige diskrete Verteilungsmodelle behandelt.

Inhaltsübersicht:

1. Einführung und Motivation
2. Eindimensionale empirische Verteilungen
3. Zweidimensionale empirische Verteilungen
4. Grundlagen der Wahrscheinlichkeitsrechnung
5. Zufallsvariablen
6. Diskrete Verteilungsmodelle

**Lernziele/Kompetenzen:**
Die Teilnehmer/-innen von „Methoden der Statistik I“ werden mit den Grundprinzipien der deskriptiven Statistik vertraut gemacht. Sie werden in die Lage versetzt, verschiedene Datentypen sicher zu unterscheiden und diese mit statistischer Software (R-Studio) zu untersuchen. Zudem werden die grundlegenden inhaltlichen Analyseverfahren bezüglich Lage, Streuung und Zusammenhängen von Merkmalen vermittelt. Zusätzlich wird den Teilnehmer/-innen ein grundlegendes Verständnis über die Regeln und Gesetzmäßigkeiten der Wahrscheinlichkeitsrechnung vermittelt.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>ab dem 1. | **Minimale Dauer des Moduls:**<br>1 Semester Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Methoden der Statistik I**<br>**Lehrformen:** Vorlesung und Übung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS | **5,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 90 Minuten<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>keine | |

| **Modul Stat-B-02 Methoden der Statistik II**<br>*Statistical Methods II* | 6 ECTS / 180 h |
|---|---|

(seit SS22)
Modulverantwortliche/r: Prof. Dr. Timo Schmid

**Inhalte:**
Die Veranstaltung „Methoden der Statistik II“ beschäftigt sich hauptsächlich mit Methoden der induktiven Statistik. Diese ermöglichen es, Rückschlüsse von einer Stichprobe auf die Grundgesamtheit zu ziehen. Als Vorbereitung werden grundlegende Konzepte und Definitionen wichtiger stetiger Verteilungsmodelle behandelt. Der Schwerpunkt liegt anschließend auf a) Schätzung eines unbekannten Parameters einer Verteilung (Punktschätzung), b) Angabe eines Vertrauensbereichs für den unbekannten Parameter (Konfidenzintervalle) und c) Aussagen über die Gleichheit bzw. Ungleichheit von Verteilungen und Parametern (Hypothesentests). Alle drei Techniken werden für die Regressionsanalyse benötigt. Die lineare Regression ist ein Verfahren zur Modellierung einfacher Modellzusammenhänge mehrerer Merkmale. Besonders die Untersuchung von Zusammenhängen ist für viele empirische Fragestellungen zentral.

Inhaltsübersicht:

Aufbauend auf der Veranstaltung „Methoden der Statistik I“:

7. Stetige Verteilungsmodelle

8. Normalverteilung

9. Stichprobenfunktionen

10. Schätzen von Parametern

11. Konfidenzintervalle

12. Testen von Hypothesen

13. Regressionsanalyse

**Lernziele/Kompetenzen:**
Die Teilnehmer/-innen von „Methoden der Statistik II“ werden mit den Grundprinzipien der induktiven Statistik vertraut gemacht. Sie werden lernen, Entscheidungen auf Basis statistischer Daten zu treffen und die Güte dieser Entscheidungen zu beurteilen. Dabei werden Sie Rückschlüsse von einer Stichprobe auf die Grundgesamtheit ziehen, statistische Hypothesen untersuchen und einfache Modellzusammenhänge berechnen. Die Teilnehmer/-innen werden in die Lage versetzt, diese Techniken anhand von empirischen Daten mit statistischer Software (R-Studio) selbständig anzuwenden.

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:**<br>Die vorherige Absolvierung des Moduls Stat-B-01 (Methoden der Statistik I). | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>ab dem 2. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Methoden der Statistik II** | **5,00 SWS** |

| | |
|---|---|
| **Lehrformen:** Vorlesung und Übung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS | |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 90 Minuten<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>keine | |

| **Modul UxD-Sem-B Bachelorseminar User Experience and Design**<br>*Bachelor Seminar User Experience and Design* | 3 ECTS / 90 h |
|---|---|
| Modulverantwortliche/r: Prof. Dr. Patrick Tobias Fischer | |

**Inhalte:**
Seminare am Lehrstuhl User Experience and Design orientieren sich an aktuellen Themen folgender wissenschaftlicher Konferenzen:

- CHI (Conference on Human Factors in Computing Systems)
- DIS (Conference on Designing Interactive Systems)
- MAB (Media Architecture Biennale Confernece)
- TEI (Conference on Tangible, Embedded and Embodied Interaction)
- MuC (Mensch und Computer)
- ...

Der Fokus eines Seminarthemas wird für eine Lehrveranstaltung jeweils neu definiert. Beispielhafte Themen sind:

- KI im Gestaltungsprozess
- Mixed Initiative / Co-Activity
- Urban HCI / Urban Interaction Design
- Values of Public Interfaces
- Untersuchung von Evaluationsmethoden für öffentliche Installationen
- Designforschung
- ...

Der Arbeitsaufwand für diese Modul gliedert sich typischerweise wie folg:

- Teilnahme an den Präsenzterminen (Themenvergabe, Besprechungen, Präsentationen): ca. 20 Stunden
- Literaturrecherche und Einarbeitung: ca. 25 Stunden
- Vorbereiten und Verfassen der Präsentation: ca. 10 Stunden
- Erstellen der schriftlichen Ausarbeitung: ca. 35 Stunden

**Lernziele/Kompetenzen:**
Vertiefung eigener Expertisen, Fachliche Anerkennung, Wissensorientierung, analytische Fähigkeiten, Sachlichkeit, Beurteilungsvermögen, Kommunikationsfähigkeit, Literaturarbeit, erkennen und angemessenes Verwenden fachlicher Konzepte, …

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:**<br>wissenschaftliches Arbeiten | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** jährlich nach Bedarf WS oder SS | **Empfohlenes Fachsemester:**<br>ab dem 2. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Seminar**<br>**Dozenten:** Prof. Dr. Patrick Tobias Fischer | **2,00 SWS** |

| | |
|---|---|
| **Sprache:** Deutsch<br>**Angebotshäufigkeit:** jährlich nach Bedarf WS oder SS | |

| | |
|---|---|
| **Prüfung**<br>Referat mit schriftl. Hausarbeit / Prüfungsdauer: 20 Minuten | |

| **Modul VIS-GIV-B Grundlagen der Informationsvisualisierung**<br>*Foundations of Information Visualization* | 6 ECTS / 180 h |
|---|---|
| (seit SS22)<br>Modulverantwortliche/r: Prof. Dr. Fabian Beck | |

**Inhalte:**
Die Veranstaltung bietet eine Einführung in die automatische Erstellung und Programmierung von interaktiven Informationsvisualisierungen, die einer explorativen Analyse und effizienten Kommunikation von Daten dienen. Dabei werden verschiedene allgemeine Ansätze zur Erstellung von Visualisierungen diskutiert und erprobt sowie zugehörige Interaktionstechniken vorgestellt. Im Zentrum der Veranstaltungen stehen universell einsetzbare Visualisierungstechniken für verschiedene abstrakte Datentypen:

- Numerische Daten (Diagramme für univariate Verteilungen, multivariate Daten und Zeitreihen)
- Kategoriale Daten (Mengen- und Ereignisvisualisierungen)
- Relationale Daten (Visualisierungen für Graphen und Hierarchien)
- Räumlich-zeitliche Daten (Visualisierung von Bewegung sowie räumlich zugeordnete Zeitreihen und Ereignisse)

Unterstützende Werkzeuge und Technologien für die Erstellung solcher Visualisierungen werden ebenfalls vorgestellt und genutzt.

**Lernziele/Kompetenzen:**
Die Studierenden kennen die wichtigsten grundlegenden Techniken der Informationsvisualisierung und können diese korrekt auf einen gegebenen Datensatz anwenden. Sie beherrschen die geometrischen Grundlagen und Algorithmen, um solche Visualisierungen eigenständig als interaktive Visualisierungen zu implementieren. Sie können entsprechende Technologien und Werkzeuge nutzen, die eine effiziente Implementierung dieser Techniken unterstützen.

**Sonstige Informationen:**
Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:

- Präsenzzeit in Vorlesung und Übung: 45h
- Vor- und Nachbereitung der Vorlesung: 30h
- Bearbeitung von Übungen und Studienleistungen: 75h
- Vorbereitung zur Prüfung: 30h

**Zulassungsvoraussetzung für die Belegung des Moduls:**
keine

| **Empfohlene Vorkenntnisse:**<br>Grundlegende Programmierkenntnisse; Algorithmen und Datenstrukturen | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:**<br>ab dem 3. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Grundlagen der Informationsvisualisierung**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Fabian Beck<br>**Sprache:** Deutsch | **2,00 SWS** |

| | |
|---|---|
| **Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>Siehe Modulbeschreibung<br>**Literatur:**<br>Weiterführende Unterlagen werden in der Veranstaltung bekannt gegeben. | |
| **2. Grundlagen der Informationsvisualisierung**<br>**Lehrformen:** Übung<br>**Dozenten:** N.N.<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich<br>**Inhalte:**<br>In der Übung werden Vorlesungsinhalte vertieft und deren praktische Anwendung geübt. | **2,00 SWS** |

| | |
|---|---|
| **Prüfung**<br>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten<br>**Beschreibung:**<br>Durch die freiwillige Abgabe von semesterbegleitenden Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Zu Beginn der Lehrveranstaltung wird bekannt gegeben, ob Studienleistungen angeboten werden. Falls Studienleistungen angeboten werden, wird zu diesem Zeitpunkt auch die Anzahl, die Art, der Umfang und die Bearbeitungsdauer der Studienleistungen sowie die Anzahl an erreichbaren Punkten pro Studienleistung und in der Modulprüfung bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden. | |

| **Modul VIS-Sem-B Bachelorseminar Informationsvisualisierung**<br>*Bachelor Seminar Information Visualization* | 3 ECTS / 90 h |
|---|---|
| (seit SS22)<br>Modulverantwortliche/r: Prof. Dr. Fabian Beck | |
| **Inhalte:**<br>Das Seminar behandelt praktische Anwendungen von Visualisierungssystemen. Es werden deren Eigenschaften diskutiert und verglichen, sowie deren Implementierung untersucht. Alle Teilnehmenden bearbeiten individuell zugewiesene Themen, die unterschiedliche Facetten zu einem übergreifenden Seminarthema beitragen. | |
| **Lernziele/Kompetenzen:**<br>Die Studierenden können ein vorgegebenes Thema der Angewandten Informatik selbstständig recherchieren. Sie lernen moderne Benutzerschnittstellen und Visualisierungssysteme zu bewerten und entwickeln ein vertieftes Verständnis des jeweiligen Themas, seiner Einsatz- und Anwendungsmöglichkeiten sowie seiner Grenzen. Sie verstehen und üben Methoden der professionellen Kommunikation in mündlicher und schriftlicher Form. | |
| **Sonstige Informationen:**<br>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:<br>• Präsenzzeit: 20h<br>• Recherche: 25h<br>• Vorbereitung der Präsentation: 15h<br>• Erstellung einer schriftlichen Ausarbeitung: 30h | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | |
| **Empfohlene Vorkenntnisse:**<br>grundlegende Programmierkenntnisse | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>ab dem 2. | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Informationsvisualisierung**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Fabian Beck, N.N.<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS<br>**Inhalte:**<br>Siehe Modulbeschreibung<br>**Literatur:**<br>Weiterführende Unterlagen werden in der Veranstaltung bekannt gegeben. | **2,00 SWS** |

| **Prüfung** | |
|---|---|

| | |
|---|---|
| Hausarbeit mit Referat / Prüfungsdauer: 20 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Die Bekanntgabe der Lehr- und Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung. | |

| **Modul WI-Seminar-B Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik**<br>*Bachelor Seminar in Information Systems* | | 3 ECTS / 90 h |
|---|---|---|
| (seit WS24/25)<br>Modulverantwortliche/r: Prof. Dr. Sven Overhage | | |
| **Inhalte:**<br>Eigenständige Erarbeitung und Präsentation eines Themas aus dem Fachgebiet der Wirtschaftsinformatik mit wissenschaftlichen Methoden. | | |
| **Lernziele/Kompetenzen:**<br>Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten. | | |
| **Sonstige Informationen:**<br>Es ist ein Bachelorseminar aus dem Fachgebiet der Wirtschaftsinformatik zu wählen. Die Seminarthemen werden über die jeweiligen Homepages der Lehrstühle bekannt gegeben. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>keine | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar**<br>**Lehrformen:** Seminar<br>**Sprache:** Deutsch/Englisch<br>**Angebotshäufigkeit:** WS, SS<br>**Literatur:**<br>Die Literatur wird zu Beginn eines Seminars bekannt gegeben. | **2,00 SWS** |

| **Prüfung** | |
|---|---|
| Hausarbeit mit Referat<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regelmäßige Teilnahme an der Lehrveranstaltung<br>**Beschreibung:**<br>Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen. Alternativ kann die Prüfungsleistung auf Hausarbeit mit Kolloquium festgelegt werden.<br>Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Referats bzw. des Kolloquiums werden zu Beginn einer jeden Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekannt gegeben. | |

| **Modul WiMa-B-002 Wirtschaftsmathematik: Analysis**<br>*Mathematics for Economics and Business: Calculus* | | 6 ECTS / 180 h |
|---|---|---|
| (seit WS22/23)<br>Modulverantwortliche/r: Prof. Dr. Anne Leucht | | |
| **Inhalte:**<br>• Funktionen in einer Variablen: Funktionsbegriff, Eigenschaften, Beispiele<br>• Grenzwerte und Stetigkeit von Funktionen in einer Variablen<br>• Differentialrechnung für Funktionen in einer Variablen: Differenzenquotient, Differentialquotient, Ableitungsregeln, Anwendung in Approximationstheorie und Optimierung, Regel von L'Hospital<br>• Funktionen mehrerer Variablen: Begriffsbildung, Beispiele, Stetigkeit, partielle Differentiation, Ableitung impliziter Funktionen, totales Differential und Anwendung in Approximationstheorie<br>• Krümmungsverhalten von Funktionen, Optimierung mit und ohne Nebenbedingungen<br>• Integrationsrechnung: Stammfunktionen, Darboux-Summen & bestimmtes Integral, unbestimmte & eigentliche Integrale, Ausblick auf Integration von Funktion in mehreren Variablen | | |
| **Lernziele/Kompetenzen:**<br>Vermittlung von mathematischen Grundkenntnissen aus dem Gebiet der Analysis. Es werden Grundlagen für das Verständnis und die Beherrschung mathematischer Formalismen, Verfahren und Konzepte geschaffen, welche in weiterführenden wirtschaftswissenschaftlichen und (wirtschafts-)informatischen Veranstaltungen zum Einsatz kommen. | | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>keine | | |
| **Empfohlene Vorkenntnisse:**<br>Inhalte der Veranstaltung Wirtschaftsmathematik: Lineare Algebra | | **Besondere Bestehensvoraussetzungen:**<br>keine |
| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:**<br>ab dem 1. | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Wirtschaftsmathematik: Analysis**<br>**Lehrformen:** Vorlesung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS<br>**Literatur:**<br>• Jensen, U. (2017), Mathematik für Wirtschaftswissenschaftler, de Gruyter.<br>• Merz, M. und Wüthrich, M. (2013), Mathematik für Wirtschaftswissenschaftler, Vahlen (München).<br>• Sydsaeter K. und Hammond, P. (2018), Mathematik für Wirtschaftswissenschaftler, Pearson (München). | **2,00 SWS** |
| **2. Übung zur Wirtschaftsmathematik: Analysis**<br>**Lehrformen:** Übung<br>**Sprache:** Deutsch<br>**Angebotshäufigkeit:** WS, SS<br>**Literatur:** | **1,00 SWS** |

| | |
|---|---|
| • Bosch, K. (2012), Übungs- und Arbeitsbuch Mathematik für Ökonomen, Oldenbourg (München).<br>• Böker, F. (2013), Mathematik für Wirtschaftswissenschaftler: das Übungsbuch, Pearson (München).<br>• Cramer, E. (2006), Vorkurs Mathematik: Arbeitsbuch zum Studienbeginn in den Wirtschafts- und Sozialwissenschaften, Springer (Berlin).<br>• Merz, M. (2013), Übungsbuch für Wirtschaftswissenschaftler, Vahlen (München).<br>• Opitz, O. et al. (2014), Mathematik-Übungsbuch: für das Studium der Wirtschaftswissenschaften, de Gruyter Oldenbourg (Berlin).<br>• Schwarze, J. (2000), Mathematik für Wirtschaftswissenschaftler / 1. Grundlagen, NWB, Verl. Neue Wirtschafts-Briefe (Herne). | |

| | |
|---|---|
| **Prüfung**<br>schriftliche Modulprüfung (Klausur) / Prüfungsdauer: 60 Minuten | |

| **Modul xAI-MML-B Mathematics for Machine Learning**<br>*Mathematics for Machine Learning* | 6 ECTS / 180 h |
|---|---|

(seit SS25)
Modulverantwortliche/r: Prof. Dr. Christian Ledig

**Inhalte:**
The course aims to establish a common mathematical foundation for the further study of advanced machine learning techniques. The content is selected specifically to be most relevant for students interested in machine learning problems and covers a broad range of concepts from, e.g., linear algebra, vector calculus, probability theory, statistics, and optimization.

**Lernziele/Kompetenzen:**
In this course students will learn fundamental mathematical concepts that are important prerequisites for the deeper understanding of the field of machine learning. The overarching goal of this course is to build a mathematical foundation by selectively covering the most essential mathematical concepts form a broad range of mathematical disciplines. Dependent on previous background, students will get the chance to learn critical ML-relevant mathematics for the first time or consolidate concepts that have been partially covered in their previous curriculum.

The lecture is accompanied by exercises and assignments that will help participants develop both theoretical and practical experience. In those exercises students will get the opportunity to learn how to apply and prove theoretical concepts as well as implement some concrete algorithms in Python and its respective commonly used libraries.

Course is also open to MSc students with the goal of building / consolidating their mathematical foundation with a focus on machine learning applications.

**Sonstige Informationen:**
The lecture is conducted in English. The workload of this module is expected to be roughly as follows:

- Lecture: 22.5h (equals the 2 SWS)
- Preparation of lectures and analysis of further sources: 30h (over the 15 weeks term)
- Exercise classes accompanying lecture: 22.5h (equals the 2 SWS)
- Work on the actual assignments: 75h (over the 15 weeks term)
- Preparation for exam: 30h

**Zulassungsvoraussetzung für die Belegung des Moduls:**
none

| **Empfohlene Vorkenntnisse:**<br>No specific prior knowledge is required. | | **Besondere Bestehensvoraussetzungen:**<br>keine |
|---|---|---|
| **Angebotshäufigkeit:** SS, jährlich | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |

| **Lehrveranstaltungen** | |
|---|---|
| **1. Mathematics for Machine Learning**<br>**Lehrformen:** Vorlesung<br>**Dozenten:** Prof. Dr. Christian Ledig<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** SS, jährlich | **2,00 SWS** |

**Lernziele:**
c.f. module description

**Inhalte:**
The lecture will be held in English. The following is a selection of topics that will be addressed in the course

- Linear Algebra (e.g., vector spaces, span, basis, rank)
- Analytic Geometry (e.g., norms, inner product, projections)
- Matrix decompositions (e.g., Eigenvectors, SVD)
- Vector calculus (e.g., derivatives, Taylor series)
- Information Theory (e.g., entropy, KL divergence)
- Probability theory and distributions
- Statistics (e.g., estimators, tests)
- Optimization (e.g., gradient based)
- Machine Learning Problems (e.g., Density estimation, Dimensionality Reduction)

**Literatur:**

- Marc. Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong: Mathematics for Machine Learning, Cambridge University Press, 2020

Further literature will be announced at the beginning of the course.

**2. Mathematics for Machine Learning** — **2,00 SWS**

**Lehrformen:** Übung
**Dozenten:** N.N.
**Sprache:** Englisch/Deutsch
**Angebotshäufigkeit:** SS, jährlich

**Lernziele:**
see module description

**Inhalte:**
Further exploration of concepts discussed in the lecture by specific assigments and some programming exercises implemented predominantly in Python.

**Literatur:**
see lecture description

**Prüfung**
schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Stunden
**Beschreibung:**
The content that is relevant for the exam consists of the content presented in the lecture and exercises/tutorials (including the assignments) as well as additional content of the discussed literature, which will be highlighted.

| **Modul xAI-Sem-B1 Bachelorseminar Erklärbares Maschinelles Lernen**<br>*Bachelor Seminar Explainable Machine Learning* | 3 ECTS / 90 h |
|---|---|
| (seit SS23)<br>Modulverantwortliche/r: Prof. Dr. Christian Ledig | |
| **Inhalte:**<br>Machine Learning holds great promise to transform a variety of industries including healthcare. However, there are key challenges when translating AI technology reliably into practice. In this seminar students will learn about a selected subarea of machine learning often in the context of a particular application. The seminar will enable students to apply knowledge from corresponding lectures and exercises and independently explore a particular research-oriented topic based on published literature. The seminar focuses on a wide spectrum of aspects not limited to pure technical questions. | |
| **Lernziele/Kompetenzen:**<br>Students will learn about the potential as well as current challenges when translating AI systems into practice. Participants will learn to independently research their specific topic by deep diving into and structuring published literature. Within the seminar students learn to present and communicate state-of-the-art research results in both oral (presentation) and written form (technical report). Seminar participants will further learn about and critically discuss scientific questions with their peers. In comparison to the Master Seminar this Bachelor Seminar is more moderate in terms of complexity of selected topics as well as expectations with respect to delivered reports and presentations. | |
| **Sonstige Informationen:**<br>This seminar is generally conducted in English. The workload of this module is expected to be roughly as follows:<br>• Attendance of seminar / presentation: 20h<br>• Literature review and familiarization with topic: 25h<br>• Preparation of presentation: 15h<br>• Written report: 30h | |
| **Zulassungsvoraussetzung für die Belegung des Moduls:**<br>none | |
| **Empfohlene Vorkenntnisse:**<br>Recommended completion of modules "Lernende System / Machine Learning" or "Einführung in die KI / Introduction into AI. | **Besondere Bestehensvoraussetzungen:**<br>keine |

| **Angebotshäufigkeit:** WS, SS | **Empfohlenes Fachsemester:** | **Minimale Dauer des Moduls:**<br>1 Semester |
|---|---|---|

| **Lehrveranstaltungen** | |
|---|---|
| **Bachelorseminar Erklärbares Maschinelles Lernen**<br>**Lehrformen:** Seminar<br>**Dozenten:** Prof. Dr. Christian Ledig<br>**Sprache:** Englisch/Deutsch<br>**Angebotshäufigkeit:** WS, SS | **2,00 SWS** |
| **Inhalte:**<br>see module description | |
| **Literatur:** | |

| | |
|---|---|
| will be announced at the beginning of the course. | |

| | |
|---|---|
| **Prüfung**<br>Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten<br>Bearbeitungsfrist: 4 Monate<br>**Zulassungsvoraussetzung zur Modulprüfung:**<br>Regular attendance of seminar and other presentations<br>**Beschreibung:**<br>The seminar will be held in English including the report and presentations. | |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A1 Fachstudium Mathematische Grundlagen** | | **30** | | |
| | Studierende, die das Modul Wirtschaftsmathematik mit 4 ECTS absolviert haben (WiMa-B-01b), erreichen in dieser Modulgruppe **28 ECTS**. | | | | |
| GdI-MfI-1 | Mathematik für Informatik 1 (Aussagen- u. Prädikatenlogik) | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| Stat-B-01 | Methoden der Statistik I | WS, SS(1) | 6 | 5 Vorlesung und Übung | schriftliche Prüfung (E-Prüfung)<br>90 Minuten |
| Stat-B-02 | Methoden der Statistik II | WS, SS(1) | 6 | 5 Vorlesung und Übung | schriftliche Prüfung (E-Prüfung)<br>90 Minuten |
| WiMa-B-002 | Wirtschaftsmathematik: Analysis | WS, SS(1) | 6 | 2 Vorlesung<br>1 Übung | schriftliche Modulprüfung (Klausur)<br>60 Minuten |
| KTR-MfI-2 | Mathematik für Informatik 2 (Lineare Algebra) | SS, jährlich | 6 | 4 Vorlesung und Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A2 Fachstudium Allgemeine Informatik Grundlagen** | | **42** | | |
| GdI-GTI-B | Grundlagen der Theoretischen Informatik | SS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| DSG-EiAPS-B | Einführung in Algorithmen, Programmierung und Software | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| AI-AuD-B | Algorithmen und Datenstrukturen | SS, jährlich(1) | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>105 Minuten |
| SWT-FPS-B | Foundations of Program Semantics | WS, jährlich | 6 | 4 Vorlesung<br>2 Übung | Hausarbeit mit Kolloquium<br>3 Wochen<br>20 Minuten |
| MOBI-DBS-B | Datenbanksysteme | WS, SS | 6 | 4 Vorlesung, Übung | schriftliche Prüfung (Klausur)<br>105 Minuten |
| PSI-EiRBS-B | Einführung in Rechner- und Betriebssysteme | SS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (E-Prüfung)<br>110 Minuten |
| PSI-IntroSP-B | Introduction to Security and Privacy | WS, jährlich(1) | 6 | 2 Vorlesung<br>2 Übung | Testat<br>90 Minuten<br>schriftliche Prüfung (E-Prüfung)<br>120 Minuten |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A3 Fachstudium Programmierung und Softwaretechnik** | | **24 - 36** | | |
| | **Pflichtbereich: A3** | | **18** | | |
| | Bis Sommersemester 2024 wurde statt GdI-MTL-B das Modul GdI-IFP-B absolviert. | | | | |
| GdI-MTL-B | Modal and Temporal Logic | WS, jährlich | 6 | 4 Vorlesung und Übung | mündliche Prüfung |
| SWT-FSE-B | Foundations of Software Engineering | SS, jährlich | 6 | 3 Vorlesung<br>3 Übung | schriftliche Prüfung (Klausur)<br>120 Minuten |
| DSG-JaP-B | Java Programmierung | WS, jährlich(2017/2018) | 3 | 2 Vorlesung, Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| DSG-AJP-B | Fortgeschrittene Java Programmierung | SS, jährlich | 3 | 2 kein Typ gewählt, Übung | Hausarbeit mit Kolloquium<br>3 Monate<br>10 Minuten |
| | **Wahlpflichtbereich: A3** | | **6 - 18** | | |
| AlgoK-AK-B | Algorithmen und Komplexität | SS, jährlich(1) | 6 | 2 Vorlesung und Übung | mündliche Prüfung |
| AlgoK-TAG | Tree decompositions, algorithms and games | WS, jährlich(1) | 6 | 4 Vorlesung und Übung | Sonstiges<br>90 Minuten |
| DT-CPP-B | Einführung in die Systemprogrammierung in C++ | WS, jährlich(1) | 6 | 4 Vorlesung und Übung | Portfolio<br>4 Monate<br>30 Minuten |
| HCI-IS-B | Interaktive Systeme | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten<br>mündliche Prüfung |
| KogSys-KI-B | Einführung in die Künstliche Intelligenz | SS, jährlich(1) | 6 | 2 Vorlesung<br>2 Übung | schriftliche Modulprüfung (Klausur)<br>105 Minuten |
| KogSys-ML-B | Einführung in Maschinelles Lernen | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>105 Minuten |
| VIS-GIV-B | Grundlagen der Informationsvisualisierung | SS, jährlich(1) | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |

| | | | | | |
|---|---|---|---|---|---|
| xAI-MML-B | Mathematics for Machine Learning | SS, jährlich(1) | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Stunden |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A4 Fachstudium Komplexe und Verteilte Systeme** | | **15 - 27** | | |
| | **Pflichtbereich: A4** | | **15** | | |
| DSG-IDistrSys-B | Introduction to Distributed Systems | SS, jährlich(2025) | 6 | 2 Vorlesung<br>2 Übung | Hausarbeit mit Kolloquium<br>3 Monate<br>10 Minuten |
| DSG-PKS-B | Programmierung komplexer interagierender Systeme | WS, jährlich | 3 | 2 Übung | Hausarbeit mit Kolloquium<br>3 Monate<br>10 Minuten |
| KTR-Datkomm-B | Datenkommunikation | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| | **Wahlpflichtbereich: A4** | | **0 - 12** | | |
| | Das Modul GdI-MTL-B ist seit Wintersemester 2024/25 hier nicht mehr wählbar. | | | | |
| HCI-US-B | Ubiquitäre Systeme | WS, jährlich | 6 | 2 Vorlesung<br>2 Übung | mündliche Prüfung<br>schriftliche Prüfung (Klausur)<br>90 Minuten |
| MI-WebT-B | Web-Technologien | SS, jährlich | 6 | 2 Vorlesung<br>2 Übung | schriftliche Prüfung (Klausur)<br>105 Minuten |
| SYSNAP-SNAP-B | Systemnahe Programmierung | WS, jährlich(WS 2026/27) | 6 | 4 Vorlesung und Übung | Portfolio<br>3 Monate<br>3 Monate |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A5 Anwendungskontext Software Systems Science** | | **8 - 16** | | |

Studierende, die das Modul Wirtschaftsmathematik mit 4 ECTS absolviert haben (WiMa-B-01b), müssen in dieser Modulgruppe **10 - 18 ECTS** erreichen.

Wählbar sind Module der Modulgruppe A3 gemäß der Studien- und Fachprüfungsordnung für den Bachelorstudiengang Angewandte Informatik und Module der Modulgruppe A1 gemäß der Studien- und Fachprüfungsordnung für den Bachelorstudiengang Wirtschaftsinformatik, sofern diese nicht in den Modulgruppen A3 und A4 wählbar sind. Ferner sind Module aus dem Nebenfachangebot gemäß Anhang APO GuK/Huwi 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.

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A6 Überfachliche Qualifikationen** | | **6 - 18** | | |
| | **Pflichtbereich: A6** | | **6** | | |
| MI-WAIAI-B | Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik | SS, jährlich(1) | 3 | 2 Vorlesung und Übung | Portfolio<br>4 Monate |
| PSI-EDS-B | Ethics for the Digital Society | WS, jährlich | 3 | 2 Vorlesung | schriftliche Prüfung (E-Prüfung)<br>80 Minuten |
| | **Wahlpflichtbereich: A6** | | **0 - 12** | | |
| | **Fach: Allgemeine Schlüsselqualifikationen** | | **0 - 12** | | |
| AIC-IITP-B | Internationales IT Projektmanagement | SS, jährlich(1) | 6 | 4 Vorlesung und Übung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| PSI-DatSchu-B | Datenschutz | SS, jährlich(1) | 3 | 2 Vorlesung | schriftliche Prüfung (Klausur)<br>90 Minuten |
| KogSys-GAI-B | Genderaspekte in der Informatik | SS, jährlich(1) | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>30 Minuten |
| | **Fach: Fremdsprachen**<br>Im Bereich Fremdsprachen können Module gemäß dem Angebot des Sprachenzentrums der Universität Bamberg absolviert werden, ausgenommen die Module der Bereiche Deutsch als Fremdsprache und Wirtschaftsdeutsch. | | **0 - 12** | | |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A7 Seminare und Projekte** | | **18 - 24** | | |
| | Es sind zusätzlich zum Pflichtbereich zwei Seminarmodule in den Themenbereichen A2-A4 mit jeweils 3 ECTS-Punkten sowie ein bis zwei Projektmodule mit je 6 ECTS-Punkten zu absolvieren. Mindestens ein Seminarmodul sowie ein Projektmodul müssen der Fächergruppe Informatik entstammen.<br><br>Für Studierende, die vor dem Wintersemester 21/22 ein Projekt mit 12 ECTS absolviert oder in Teilen absolviert haben, ersetzt dieses zwei Projektmodule.<br><br>Die Zulassung zur Modulprüfung setzt bei den Modulen dieser Modulgruppe eine regelmäßige Teilnahme an den zugehörigen Lehrveranstaltungen gem. § 9 Abs. 10 APO voraus. | | | | |
| | **Pflichtbereich: Pflichtbereich** | | **6** | | |
| SWT-SWL-B | Software Engineering Lab | WS, jährlich | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium (schriftliche Hausarbeit mit Kolloquium)<br>2 Wochen<br>45 Minuten |
| | **Wahlpflichtbereich: Seminare** | | **6** | | |
| | **Fach: Seminar(e) in Informatik** | | **3 - 6** | | |
| AlgoK-Sem-B | Bachelorseminar Algorithmen und Komplexitätstheorie | jährlich nach Bedarf WS und SS(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>4 Monate<br>30 Minuten |
| DSG-Sem-B | Bachelorseminar zur Praktischen Informatik | WS, SS | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>4 Monate<br>20 Minuten |
| DT-DB4MLKD-B | Moderne Datenbanksysteme für maschinelles Lernen und Wissensentdeckung | jährlich nach Bedarf WS und SS(1) | 3 | 3 Seminar | Referat mit schriftl. Hausarbeit<br>14 Tage<br>30 Minuten |
| GdI-Sem-B | Bachelorseminar Grundlagen der Informatik | jährlich nach Bedarf WS und SS | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>30 Minuten |

| | | | | | |
|---|---|---|---|---|---|
| KTR-Sem-B | Bachelorseminar zu Kommunikationssystemen und Rechnernetzen | jährlich nach Bedarf WS und SS(Regelturnus: SS) | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>30 Minuten |
| MOBI-SEM-B | Bachelor-Seminar Mobile Software Systems | WS, jährlich(1) | 3 | 2 Seminar | Hausarbeit mit Referat |
| PSI-Sem-B | Seminar Security and Privacy Foundations | WS, SS(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>2 Monate<br>30 Minuten |
| SWT-SEM-B | Seminar Software Engineering and Programming Languages (Bachelor) | SS, jährlich | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>8 Wochen<br>40 Minuten |
| SYSNAP-SEM-B | Seminar System Software | WS, SS(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>4 Monate<br>30 Minuten |
| | **Fach: Seminar in Angewandter Informatik oder Wirtschaftsinformatik** | | **0 - 3** | | |
| AISE-SemCP-B | Bachelorseminar Computational Philosophy | WS, jährlich(1) | 3 | 2 Seminar | Hausarbeit mit Referat<br>3 Monate<br>30 Minuten |
| AISE-SemSC-B | Bachelorseminar Smart City | SS, jährlich(1) | 3 | 2 Blockseminar | Hausarbeit mit Referat |
| CG-SemCGA-B | Seminar Computergrafik und Animation | SS, jährlich(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>2 Monate<br>20 Minuten |
| CG-SemVRAR-B | Seminar Virtual Reality / Augmented Reality | WS, jährlich(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>2 Monate<br>20 Minuten |
| DS-Sem-B | Bachelorseminar Dialogsysteme | WS, SS(1) | 3 | 2 Seminar | Hausarbeit mit Referat |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | 15 Minuten |
| HCI-Sem-B | Bachelorseminar Mensch-Computer-Interaktion | SS, jährlich | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>4 Monate<br>30 Minuten |
| KInf-Seminar-B | Bachelorseminar Kulturinformatik | SS, jährlich | 3 | 2 Seminar | Hausarbeit mit Referat<br>20 Minuten<br>4 Monate |
| KogSys-Sem-B | Bachelorseminar Kognitive Systeme | WS, jährlich | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>30 Minuten |
| MI-Sem-B | Bachelorseminar zur Medieninformatik | SS, jährlich | 3 | 2 Proseminar | Hausarbeit mit Referat<br>30 Minuten |
| MII-SemHRI-B | Bachelorseminar Mensch-Roboter-Interaktion | WS, jährlich(1) | 3 | 2 Seminar | Referat mit schriftl. Hausarbeit<br>4 Monate<br>20 Minuten |
| UxD-Sem-B | Bachelorseminar User Experience and Design | jährlich nach Bedarf WS oder SS(1) | 3 | 2 | Referat mit schriftl. Hausarbeit<br>20 Minuten |
| VIS-Sem-B | Bachelorseminar Informationsvisualisierung | WS, SS(1) | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>20 Minuten |
| WI-Seminar-B | Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik | WS, SS | 3 | 2 Seminar | Hausarbeit mit Referat |
| xAI-Sem-B1 | Bachelorseminar Erklärbares Maschinelles Lernen | WS, SS(1) | 3 | 2 Seminar | Hausarbeit mit Referat<br>4 Monate<br>30 Minuten |
| | **Wahlpflichtbereich: Projekte** | | **6 - 12** | | |
| DSG-Project-B | Bachelorprojekt zur Praktischen Informatik | SS, jährlich | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium<br>2 Monate<br>10 Minuten |
| DT-Proj-B | Bachelor-Projekt: Data Engineering | WS, SS(1) | 6 | 4 Projekt | Kolloquium, schr. Hausarbeit |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | 3 Monate<br>30 Minuten |
| GdI-Proj-B | Bachelorprojekt Grundlagen der Informatik | WS, SS | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium<br>4 Monate<br>20 Minuten |
| KTR-Proj | Projekt Kommunikationsnetze und -dienste | WS, jährlich(nach Bedarf auch SS) | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium<br>4 Monate<br>30 Minuten |
| MOBI-Proj-B | Bachelor Project Mobile Software Systems | WS, SS(1) | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium |
| PSI-ProjectPAD | Project Practical Attacks and Defenses | WS, SS(1) | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium<br>3 Monate<br>30 Minuten |
| SWT-SWP-B | Software Engineering Project | SS, jährlich(1) | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium (Software Engineering Project (schriftliche Hausarbeit mit Kolloquium))<br>12 Wochen<br>20 Minuten |
| SYSNAP-Project-B | Projekt Systemnahe Programmierung | WS, SS(1) | 6 | 4 Projektseminar | Hausarbeit mit Kolloquium<br>3 Monate<br>30 Minuten |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **A8 Bachelorarbeit** | | **12** | | |
| SoSySc-Thesis-B | Bachelorarbeit in Software Systems Science | WS, SS(1) | 12 | | schriftliche Hausarbeit<br>4 Monate |

| ID | Modul | Semester | ECTS | SWS | Prüfung |
|---|---|---|---|---|---|
| | **Module gemäß Abweichungen im Modulangebot A1 & A2** | | | | |
| | Um die Module<br>• GdI-GTI-B in A2<br>• DSG-EiAPS-B in A2<br>• PSI-EiRBS-B in A2<br>• GdI-MfI-1 in A1<br>abzuschließen, muss auf die Lehrveranstaltungen zu folgenden Modulen zurückgegriffen werden (Reihenfolge entspricht Auflistung): | | | | |
| Inf-DM-B | Diskrete Modellierung | WS, jährlich(1) | 9 | 6 Vorlesung und Übung | schriftliche Prüfung (Klausur) |
| Inf-Einf-B | Einführung in die Informatik | WS, jährlich(1) | 9 | 4 Vorlesung<br>2 Übung | schriftliche Prüfung (E-Prüfung)<br>180 Minuten |
| Inf-GRABS-B | Grundlagen der Rechnerarchitektur und Betriebssysteme | SS, jährlich(1) | 9 | 4 Vorlesung<br>2 Praktikum, Übung | schriftliche Prüfung (E-Prüfung)<br>90 Minuten<br>90 Minuten |
| Inf-LBR-B | Logik und Berechenbarkeit | SS, jährlich(SS 2025) | 9 | 6 Vorlesung und Übung | schriftliche Prüfung (Klausur)<br>135 Minuten |