



Otto-Friedrich Universität Bamberg

---

# **Modulhandbuch**

## **Bachelorstudiengang Angewandte Informatik**

### **Wirtschaftsinformatik und Angewandte Informatik**

**Gemäß der geltenden Fassung der Studien- und Fachprüfungsordnung vom 24.09.2024 für den Bachelorstudiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg. Gültig ab Wintersemester 2025/26.**

---

## **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**

- 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.

## Äquivalenzliste BSc Angewandte Informatik,

StuFPO vom 24.09.2024

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)
MI-WebT-B	Web-Technologien	SS25	AI-WebT-B	Web-Technologien	WS25/26
SWT-SWL-B	Software Engineering Lab	SS25	ESE-SEL-B	Software Engineering Lab	WS25/26
DSG-JaP-B	Java Programmierung	SS25	GAMES-Java-B	Objektorientierte Programmierung mit Java	WS25/26



---

# Module

AI-AuD-B: Algorithmen und Datenstrukturen.....	11
AI-Einf-B: Einführung in die Angewandte Informatik.....	14
AI-Projekt1-B: Bachelorprojekt 1 der Fächergruppe Angewandte Informatik.....	15
AI-Projekt2-B: Bachelorprojekt 2 der Fächergruppe Angewandte Informatik.....	17
AI-Projekt3-B: Bachelorprojekt 3 der Fächergruppe Angewandte Informatik.....	19
AI-Seminar1-B: Bachelorseminar 1 der Fächergruppe Angewandte Informatik.....	21
AI-Seminar2-B: Bachelorseminar 2 der Fächergruppe Angewandte Informatik.....	23
AI-Thesis-B: Bachelorarbeit im Studiengang Angewandte Informatik.....	25
AI-WebT-B: Web-Technologien.....	26
AIC-IITP-B: Internationales IT Projektmanagement.....	28
AISE-DO-B: DevOps für KI-Systeme.....	30
AISE-LKR-B: Logische Wissensrepräsentation und Schließen.....	32
AlgoK-AK-B: Algorithmen und Komplexität.....	34
AlgoK-ALDAI-B: Algorithms and logic in data science and AI.....	37
CG-CGA-B: Computergrafik und Animation.....	39
COMNET-RN-B: Rechnernetze.....	41
DS-IDS-B: Einführung in die Dialogsysteme.....	43
DT-CPP-B: Einführung in die Systemprogrammierung in C++.....	45
EESYS-SaD-B: Statistik und Data Science.....	46
ESE-SEL-B: Software Engineering Lab.....	50
ESE-SRE-B: Software Requirements Engineering.....	52
GAMES-Java-B: Objektorientierte Programmierung mit Java.....	54
Gdl-MTL-B: Modal and Temporal Logic.....	56
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis.....	58
HCI-IS-B: Interaktive Systeme.....	60
HCI-KS-B: Kooperative Systeme.....	63
HCI-US-B: Ubiquitäre Systeme.....	66
Inf-DM-B: Diskrete Modellierung.....	69
Inf-Einf-B: Einführung in die Informatik.....	71

Inf-GRABS-B: Grundlagen der Rechnerarchitektur und Betriebssysteme.....	74
Inf-LBR-B: Logik und Berechenbarkeit.....	76
Inf-Ment-B: Studieneinstiegsmentoring für Bachelorstudierende.....	80
Inf-Prog-C-B: Einführung in die C-Programmierung.....	82
Inf-Projekt1-B: Bachelorprojekt 1 der Fächergruppe Informatik.....	84
Inf-Seminar1-B: Bachelorseminar 1 der Fächergruppe Informatik.....	86
KogSys-GAI-B: Genderaspekte in der Informatik.....	88
KogSys-KI-B: Einführung in die Künstliche Intelligenz.....	90
KogSys-ML-B: Einführung in Maschinelles Lernen.....	93
MI-EMI-B: Einführung in die Medieninformatik.....	96
MI-WAIAI-B: Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik.....	99
MII-ROB-B: Einführung in die Robotik.....	101
MOBI-DBS-B: Datenbanksysteme.....	103
MOBI-DE-B: Data Engineering.....	105
NLProc-ALV-B: Algorithmisches Sprachverstehen.....	106
NLProc-IRTM-B: Information Retrieval and Text Mining.....	108
PSI-DatSchu-B: Datenschutz.....	110
PSI-EDS-B: Ethics for the Digital Society.....	111
PSI-IntroSP-B: Introduction to Security and Privacy.....	113
SWT-FSE-B: Foundations of Software Engineering.....	116
SYSNAP-SNAP-B: Systemnahe Programmierung.....	118
SuStat-013-M: Introduction to Econometrics.....	120
SuStat-014-M: Advanced Econometrics.....	121
SuStat-071-M: Advanced Data Analysis With R.....	122
SuStat-075-M: Statistische Programmierung mit R.....	124
SuStat-076-M: Fortgeschrittene Statistik.....	125
SuStat-079-M: Analyse hochdimensionaler Daten.....	126
UxD-G-M: Grundlagen des Gestaltens.....	127
VIS-GIV-B: Grundlagen der Informationsvisualisierung.....	131
WI-Projekt-B: Bachelorprojekt aus der Fächergruppe Wirtschaftsinformatik.....	133
WI-Seminar-B: Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik.....	134

WiMa-B-001: Wirtschaftsmathematik: Lineare Algebra..... 135  
WiMa-B-002: Wirtschaftsmathematik: Analysis.....137  
xAI-MML-B: Mathematics for Machine Learning.....139

# Übersicht nach Modulgruppen

## 1) A1 Fachstudium Mathematische Grundlagen (Modulgruppe) ECTS: 21

EESYS-SaD-B: Statistik und Data Science (9 ECTS, SS, jährlich).....	46
WiMa-B-001: Wirtschaftsmathematik: Lineare Algebra (6 ECTS, WS, SS).....	135
WiMa-B-002: Wirtschaftsmathematik: Analysis (6 ECTS, WS, SS).....	137

## 2) A2 Fachstudium Informatik (Modulgruppe) ECTS: 51 - 63

### a) A2 (Pflichtbereich) ECTS: 51

AI-AuD-B: Algorithmen und Datenstrukturen (6 ECTS, SS, jährlich).....	11
Inf-DM-B: Diskrete Modellierung (9 ECTS, WS, jährlich).....	69
Inf-Einf-B: Einführung in die Informatik (9 ECTS, WS, jährlich).....	71
Inf-GRABS-B: Grundlagen der Rechnerarchitektur und Betriebssysteme (9 ECTS, SS, jährlich).....	74
MOBI-DBS-B: Datenbanksysteme (6 ECTS, WS, SS).....	103
PSI-IntroSP-B: Introduction to Security and Privacy (6 ECTS, WS, jährlich).....	113
SWT-FSE-B: Foundations of Software Engineering (6 ECTS, SS, jährlich).....	116

### b) A2 (Wahlpflichtbereich) ECTS: 0 - 12

AlgoK-AK-B: Algorithmen und Komplexität (6 ECTS, SS, jährlich).....	34
AlgoK-ALDAI-B: Algorithms and logic in data science and AI (6 ECTS, WS, jährlich).....	37
COMNET-RN-B: Rechnernetze (6 ECTS, WS, jährlich).....	41
DT-CPP-B: Einführung in die Systemprogrammierung in C++ (6 ECTS, WS, jährlich).....	45
ESE-SRE-B: Software Requirements Engineering (6 ECTS, SS, jährlich).....	52
GAMES-Java-B: Objektorientierte Programmierung mit Java (3 ECTS, WS, jährlich).....	54
Gdl-MTL-B: Modal and Temporal Logic (6 ECTS, WS, jährlich).....	56
Inf-LBR-B: Logik und Berechenbarkeit (9 ECTS, SS, jährlich).....	76
Inf-Prog-C-B: Einführung in die C-Programmierung (3 ECTS, SS, jährlich).....	82
MOBI-DE-B: Data Engineering (6 ECTS, SS, jährlich).....	105
SYSNAP-SNAP-B: Systemnahe Programmierung (6 ECTS, WS, jährlich).....	118

## 3) A3 Fachstudium Angewandte Informatik (Modulgruppe) ECTS: 39 - 51

**a) A3 (Pflichtbereich) ECTS: 9**

AI-Einf-B: Einführung in die Angewandte Informatik (3 ECTS, WS, jährlich).....	14
KogSys-KI-B: Einführung in die Künstliche Intelligenz (6 ECTS, SS, jährlich).....	90

**b) A3 (Wahlpflichtbereich) ECTS: 30 - 42**

AI-WebT-B: Web-Technologien (6 ECTS, SS, jährlich).....	26
AISE-DO-B: DevOps für KI-Systeme (6 ECTS, WS, jährlich).....	30
AISE-LKR-B: Logische Wissensrepräsentation und Schließen (6 ECTS, WS, jährlich).....	32
CG-CGA-B: Computergrafik und Animation (6 ECTS, WS, jährlich).....	39
DS-IDS-B: Einführung in die Dialogsysteme (6 ECTS, WS, jährlich).....	43
HCI-IS-B: Interaktive Systeme (6 ECTS, WS, jährlich).....	60
HCI-KS-B: Kooperative Systeme (6 ECTS, SS, jährlich).....	63
HCI-US-B: Ubiquitäre Systeme (6 ECTS, WS, jährlich).....	66
KogSys-ML-B: Einführung in Maschinelles Lernen (6 ECTS, WS, jährlich).....	93
MI-EMI-B: Einführung in die Medieninformatik (6 ECTS, WS, jährlich).....	96
MII-ROB-B: Einführung in die Robotik (6 ECTS, WS, jährlich).....	101
NLProc-ALV-B: Algorithmisches Sprachverstehen (6 ECTS, WS, jährlich).....	106
NLProc-IRTM-B: Information Retrieval and Text Mining (6 ECTS, SS, jährlich).....	108
UxD-G-M: Grundlagen des Gestaltens (6 ECTS, SS, jährlich).....	127
VIS-GIV-B: Grundlagen der Informationsvisualisierung (6 ECTS, SS, jährlich).....	131
xAI-MML-B: Mathematics for Machine Learning (6 ECTS, SS, jährlich).....	139

**4) A4 Anwendungskontext Angewandte Informatik (Modulgruppe) ECTS: 18 - 30**

In der Modulgruppe A4 sind Module im Umfang von insgesamt 18 bis 30 ECTS#Punkten zu absolvieren. Die Module dürfen nicht aus dem Modulangebot der Fakultät WIAI stammen oder diesem Modulangebot gleichwertig sein. Es können Module eines oder mehrerer anderer Fächer studiert werden. Beim Studium mehrerer Fächer müssen in zwei verschiedenen Fächern jeweils mindestens 9 ECTS-Punkte erbracht werden.

Module wirtschaftswissenschaftlicher Fächer können im Umfang von bis zu 18 ECTS#Punkten eingebracht werden.

Es sind unter Berücksichtigung der oben genannten Einschränkungen beispielsweise Module aus dem Nebenfachangebot der 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. Ergibt sich aufgrund der Modulvorgaben des Anwendungsfachs ein Mindestvolumen, welches

die genannte Minimalanforderung von 9 ECTS-Punkten überschreitet, z.B. 15 ECTS, so ist diese Vorgabe zu berücksichtigen. Die von der anbietenden Fakultät in der Studien- und Fachprüfungsordnung und im jeweiligen Modulhandbuch festgelegten Regelungen zur Anmeldung und Anrechnung von Modulteilprüfungen einzelner Module und ihrer zugeordneten Lehrveranstaltungen sind bindend. Im Hinblick auf einzelne Anwendungsfächer gelten 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.unibamberg.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.

Ferner ist das Anwendungsfach "Statistik und Ökonometrie" mit den unten gelisteten Modulen einbringbar. Dieses wird nicht den Wirtschaftswissenschaften zugerechnet.

Für Module der Universität Bamberg aus dem Fach Psychologie gilt zudem Folgendes:

- Zusätzlich zum Pflichtmodul „Einführung in die Psychologie, ihre Geschichte und ethische Grundlagen für (Angewandte) Informatik und IRD“ können ein bis zwei Wahlpflichtmodule im Umfang von je 6 ECTS#Punkten eingebracht werden.
- Es stehen folgende Wahlpflichtmodule zur Auswahl:
  1. Arbeits- und Organisationspsychologie für (Angewandte) Informatik und IRD
  2. Allgemeine Psychologie I für (Angewandte) Informatik und IRD
  3. Allgemeine Psychologie II für (Angewandte) Informatik und IRD
  4. Biologische Psychologie für (Angewandte) Informatik und IRD
  5. Persönlichkeitspsychologie für (Angewandte) Informatik und IRD
  6. Sozialpsychologie für (Angewandte) Informatik und IRD.
- Die Modulprüfung wird jeweils durch schriftliche Prüfung (90 Minuten) oder mündliche Prüfung (15 Minuten) erbracht.
- Das konkrete Angebot der aus dem Fach Psychologie wählbaren Module, sowie die konkreten Modulbeschreibungen sind dem Modulhandbuch für Module des Fachs Psychologie, die im Rahmen des Bachelor- und des Masterstudiengangs Angewandte Informatik, des Bachelorstudiengangs Informatik sowie des Masterstudiengangs Interaction Research & Design erbracht werden können, zu entnehmen.

## **a) Anwendungsfach Statistik und Ökonometrie (Wahlpflichtbereich) ECTS: 0 - 30**

Die Module Statistische Programmierung mit R und Advanced Data Analysis with R unterliegen dem Vorbehalt von freien Kapazitäten im CIP-Pool.

SuStat-013-M: Introduction to Econometrics (6 ECTS, WS, jährlich).....	120
SuStat-014-M: Advanced Econometrics (6 ECTS, SS, jährlich).....	121
SuStat-071-M: Advanced Data Analysis With R (6 ECTS, SS, jährlich).....	122
SuStat-075-M: Statistische Programmierung mit R (6 ECTS, WS, SS).....	124
SuStat-076-M: Fortgeschrittene Statistik (6 ECTS, WS, jährlich).....	125
SuStat-079-M: Analyse hochdimensionaler Daten (6 ECTS, SS, jährlich).....	126

## 5) A5 Überfachliche Qualifikationen (Modulgruppe) ECTS: 7 - 13

### a) A5 (Pflichtbereich) ECTS: 7

Inf-Ment-B: Studieneinstiegsmentoring für Bachelorstudierende (1 ECTS, WS, SS).....	80
MI-WAIAI-B: Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik (3 ECTS, SS, jährlich).....	99
PSI-EDS-B: Ethics for the Digital Society (3 ECTS, WS, jährlich).....	111

### b) A5 (Wahlpflichtbereich) ECTS: 0 - 6

Im Wahlpflichtbereich kann frei aus den Bereichen Fremdsprachen, Philosophie/Ethik und Allgemeine Schlüsselqualifikationen gewählt werden.

- Im Bereich Fremdsprachen können Module gemäß dem Angebot des Sprachenzentrums Bamberg absolviert werden. Ausgenommen sind Module in der Sprache, in der die Hochschulzugangsberechtigung erworben wurde. Ausgenommen sind darüber hinaus Module zu Fremdsprachen bis zu dem Niveau, das in der Hochschulzugangsberechtigung ausgewiesen wurde. Dies gilt nicht für die Module der Bereiche der Wirtschaftsfremdsprachen und des Bereichs IT-English. Die Module der Bereiche Deutsch als Fremdsprache und Wirtschaftsdeutsch sind ab der Niveaustufe C1 wählbar, sofern die Sprache, in der Hochschulzugangsberechtigung erworben wurde, nicht Deutsch war. Einzelheiten, insbesondere die zur Auswahl stehenden Module sowie die jeweils abzulegenden Modulprüfungen und Modulteilprüfungen, sind in der Prüfungsordnung und dem Modulhandbuch für sprachpraktische Module der Otto-Friedrich-Universität Bamberg festgelegt.
- Im Bereich Philosophie/Ethik sind auf Antrag Module wählbar, die der Ethik oder der Philosophie zuzuordnen sind und im Studium Generale angeboten werden.
- Im Bereich Allgemeine Schlüsselqualifikationen stehen neben Modulen des Zentrums für Schlüsselkompetenzen gemäß Studien- und Fachprüfungsordnung für Module und Zertifikate im Bereich der Schlüsselkompetenzen § 4 Abs 2 folgende Module zur Auswahl:

AIC-IITP-B: Internationales IT Projektmanagement (6 ECTS, SS, jährlich).....	28
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis (6 ECTS, SS, jährlich).....	58
KogSys-GAI-B: Genderaspekte in der Informatik (3 ECTS, SS, jährlich).....	88
PSI-DatSchu-B: Datenschutz (3 ECTS, SS, jährlich).....	110

## 6) A6 Seminare und Projekte (Modulgruppe) ECTS: 18 - 24

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

#### aa) Seminare der Angewandten Informatik (Fach) ECTS: 3 - 6

AI-Seminar1-B: Bachelorseminar 1 der Fächergruppe Angewandte Informatik (3 ECTS, WS, SS).....	21
AI-Seminar2-B: Bachelorseminar 2 der Fächergruppe Angewandte Informatik (3 ECTS, WS, SS).....	23

**bb) Seminare der Informatik oder Wirtschaftsinformatik (Fach) ECTS: 0 - 3**

Inf-Seminar1-B: Bachelorseminar 1 der Fächergruppe Informatik (3 ECTS, WS, SS)..... 86  
WI-Seminar-B: Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik (3 ECTS, WS, SS).....134

**b) Projekte (Wahlpflichtbereich) ECTS: 12 - 18**

**aa) Projekte der Angewandten Informatik (Fach) ECTS: 12 - 18**

AI-Projekt1-B: Bachelorprojekt 1 der Fächergruppe Angewandte Informatik (6 ECTS, WS, SS)..... 15  
AI-Projekt2-B: Bachelorprojekt 2 der Fächergruppe Angewandte Informatik (6 ECTS, WS, SS)..... 17  
AI-Projekt3-B: Bachelorprojekt 3 der Fächergruppe Angewandte Informatik (6 ECTS, WS, SS)..... 19

**bb) Projekte der Informatik oder Wirtschaftsinformatik (Fach) ECTS: 0 - 6**

ESE-SEL-B: Software Engineering Lab (6 ECTS, )..... 50  
Inf-Projekt1-B: Bachelorprojekt 1 der Fächergruppe Informatik (6 ECTS, WS, SS)..... 84  
WI-Projekt-B: Bachelorprojekt aus der Fächergruppe Wirtschaftsinformatik (6 ECTS, WS, SS)..... 133

**7) A7 Bachelorarbeit (Modulgruppe) ECTS: 12**

AI-Thesis-B: Bachelorarbeit im Studiengang Angewandte Informatik (12 ECTS, WS, SS)..... 25

<b>Modul AI-AuD-B Algorithmen und Datenstrukturen</b> <i>Algorithms and Data Structures</i>		6 ECTS / 180 h 42 h Präsenzzeit 138 h Selbststudium
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> 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: <ul style="list-style-type: none"> <li>• Vorlesung 21h (14 Wochen à 1,5 Stunden)</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30h</li> <li>• semesterbegleitendes Üben und Bearbeiten von Übungsaufgaben und Teilleistungen: ca. 80h</li> <li>• Übung/Tutorium 21h (14 Wochen à 1,5 Stunden)</li> <li>• Klausur sowie Klausurvorbereitung basierend auf dem erarbeiteten Stoff: ca. 30h</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> 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.  Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

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

<p>Die Vorlesung betrachtet die zentralen Bereiche des Themengebietes Algorithmen und Datenstrukturen:</p> <ul style="list-style-type: none"> <li>• Komplexitätsbegriffe (insb. Laufzeitkomplexität, Speicherplatzkomplexität, O-Notation)</li> <li>• Korrektheit von Algorithmen</li> <li>• Listen (einfach/doppelt verkettet, Stack, Queue)</li> <li>• Hashverfahren</li> <li>• Bäume (Datenstruktur, Traversierung, Binär-, AVL-, Suchbäume, Heap)</li> <li>• Graphen (Datenstruktur, DFS-, BFS-, Dijkstra-Algorithmus, grundlegende graphentheoretische Konzepte)</li> <li>• Sortieren</li> <li>• Algorithmenkonstruktion</li> </ul>	
<p><b>Literatur:</b> Als begleitende Lektüre wird ein Standardlehrbuch über Algorithmen und Datenstrukturen empfohlen, zum Beispiel:</p> <ul style="list-style-type: none"> <li>• Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest und Clifford Stein. Introduction to Algorithms, 4. Aufl., MIT Press, 2022</li> <li>• Guter Saake und Kai-Uwe Sattler Algorithmen und Datenstrukturen: Eine Einführung mit JAVA, ISBN: 978-3864901362, 5. Aufl. 2013, 576 Seiten, dpunkt.lehrbuch</li> <li>• Thomas Ottmann und Peter Widmayer: Algorithmen und Datenstrukturen, ISBN: 978-3827428035, 5. Aufl. 2012, 800 Seiten, Spektrum, Akademischer Verlag</li> </ul>	
<p><b>2. Algorithmen und Datenstrukturen</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Medieninformatik <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> In der Übung werden Vorlesungsinhalte vertieft und deren praktische Anwendung geübt. Insbesondere werden folgende Aspekte betrachtet:</p> <ul style="list-style-type: none"> <li>• Verständnis und Nutzung von Algorithmen</li> <li>• Aufwandsbestimmung für Algorithmen</li> <li>• Implementierung von Algorithmen und Datenstrukturen</li> <li>• abstrakte Datentypen sowie Nutzung von Bibliotheken</li> <li>• Anwendung von Prinzipien zur Algorithmenkonstruktion</li> </ul> <hr/> <p><b>Literatur:</b> siehe Vorlesung; weitere Literaturempfehlungen werden in der Übung bekanntgegeben</p>	<p><b>2,00 SWS</b></p>

<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten <b>Beschreibung:</b></p>	
--	--

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.

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.

<b>Modul AI-Einf-B Einführung in die Angewandte Informatik</b> <i>Introduction to Applied Computer Science</i>		3 ECTS / 90 h 22 h Präsenzzeit 68 h Selbststudium
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Im Modul wird in Themenfelder, Methoden und Herausforderungen der Angewandten Informatik eingeführt. Dazu werden Vorlesungen, Fallstudien und Gastvorträge zu aktuellen Themen genutzt.		
<b>Lernziele/Kompetenzen:</b> Studierende erhalten einen Überblick über Anwendungsfelder der Angewandten Informatik und die besonderen Herausforderungen in exemplarischen Anwendungsfeldern. Sie lernen das methodische Vorgehen in der Angewandten Informatik kennen. Die Studierenden werden durch das Modul in die Lage versetzt, die Ziele ihres Studiums klar zu definieren und die Wahlmöglichkeiten insbesondere in der Angewandten Informatik und im Hinblick auf den zu vertiefenden Anwendungskontext zielgerichtet zu nutzen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Das Modul richtet sich ausdrücklich an Studierende im 1. und 2. Semester. Es sind keine Vorkenntnisse nötig.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Einführung in die Angewandte Informatik</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Andreas Henrich <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> Die Lernziele entsprechen den oben angegebenen Lernzielen des Moduls insgesamt.	

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Gegenstand der Klausur sind alle Inhalte von Vorlesung und Übung. Im Semester werden studienbegleitend (wöchentlich oder zweiwöchentlich) Aufgabenstellungen ausgegeben und besprochen, die ebenfalls relevant für die Klausur sind.	
---	--

<b>Modul AI-Projekt1-B Bachelorprojekt 1 der Fächergruppe Angewandte Informatik</b> <i>Bachelor Project 1 in Applied Computer Science</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Angewandte Informatik)		
<b>Inhalte:</b> Das Modul behandelt die praktische Anwendung grundlegender Methoden aus dem Bereich der Angewandten Informatik im Rahmen eines Projektes. Die behandelten Problemstellungen stammen aus den typischen Anwendungsfeldern der Angewandten Informatik.		
<b>Lernziele/Kompetenzen:</b> I.d.R aufbauend auf den in den Vorlesungen und Übungen des jeweiligen Faches erworbenen Kenntnissen und Fertigkeiten wird im Projekt eine Forschungs- und/oder Entwicklungsaufgabe mit wissenschaftlichem Bezug in einer Gruppe bearbeitet. Dabei werden die Fähigkeiten im Bereich der Systemanalyse und -entwicklung ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und in der Gruppenarbeit.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorprojekt aus einem der Fachgebiete der Angewandten Informatik zu wählen. Die wählbaren Projekte sind in UnivIS über das Schlagwort "AI-Projekt" auffindbar und als Projekte für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Bachelorprojekt 1</b> <b>Lehrformen:</b> Projektseminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS		<b>4,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben		
<b>Inhalte:</b> Die Inhalte der Bachelorprojekte werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		
<b>Literatur:</b> Die Literatur wird zu Beginn eines Projekts von jedem anbietenden Fachgebiet bekannt gegeben.		

**Prüfung**

Hausarbeit mit Kolloquium

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen.

Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden von der Betreuerin bzw. dem Betreuer des Projekts zu Beginn der Lehrveranstaltung bekannt gegeben.

<b>Modul AI-Projekt2-B Bachelorprojekt 2 der Fächergruppe Angewandte Informatik</b> <i>Bachelor Project 2 in Applied Computer Science</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Angewandte Informatik)		
<b>Inhalte:</b> Das Modul behandelt die praktische Anwendung grundlegender Methoden aus dem Bereich der Angewandten Informatik im Rahmen eines Projektes. Die behandelten Problemstellungen stammen aus den typischen Anwendungsfeldern der Angewandten Informatik.		
<b>Lernziele/Kompetenzen:</b> I.d.R aufbauend auf den in den Vorlesungen und Übungen des jeweiligen Faches erworbenen Kenntnissen und Fertigkeiten wird im Projekt eine Forschungs- und/oder Entwicklungsaufgabe mit wissenschaftlichem Bezug in einer Gruppe bearbeitet. Dabei werden die Fähigkeiten im Bereich der Systemanalyse und -entwicklung ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und in der Gruppenarbeit.  In Relation zum Bachelorprojekt 1 der Fächergruppe Angewandte Informatik werden dabei die Kompetenzen durch die Bearbeitung eines weiteren Themas verbreitert und vertieft.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorprojekt aus einem der Fachgebiete der Angewandten Informatik zu wählen. Die wählbaren Projekte sind in UnivIS über das Schlagwort "AI-Projekt" auffindbar und als Projekte für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Bachelorprojekt 2</b> <b>Lehrformen:</b> Projektseminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS		<b>4,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben		
<b>Inhalte:</b> Die Inhalte der Bachelorprojekte werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		
<b>Literatur:</b>		

Die Literatur wird zu Beginn eines Projekts von jedem anbietenden Fachgebiet bekannt gegeben.	
---	--

<b>Prüfung</b>	
----------------	--

Hausarbeit mit Kolloquium

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen.

Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden von der Betreuerin bzw. dem Betreuer des Projekts zu Beginn der Lehrveranstaltung bekannt gegeben.

<b>Modul AI-Projekt3-B Bachelorprojekt 3 der Fächergruppe Angewandte Informatik</b> <i>Bachelor Project 3 in Applied Computer Science</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Angewandte Informatik)		
<b>Inhalte:</b> Das Modul behandelt die praktische Anwendung grundlegender Methoden aus dem Bereich der Angewandten Informatik im Rahmen eines Projektes. Die behandelten Problemstellungen stammen aus den typischen Anwendungsfeldern der Angewandten Informatik.		
<b>Lernziele/Kompetenzen:</b> I.d.R aufbauend auf den in den Vorlesungen und Übungen des jeweiligen Faches erworbenen Kenntnissen und Fertigkeiten wird im Projekt eine Forschungs- und/oder Entwicklungsaufgabe mit wissenschaftlichem Bezug in einer Gruppe bearbeitet. Dabei werden die Fähigkeiten im Bereich der Systemanalyse und -entwicklung ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und in der Gruppenarbeit.  In Relation zu den Bachelorprojekten 1 und 2 der Fächergruppe Angewandte Informatik werden dabei die Kompetenzen durch die Bearbeitung eines weiteren Themas verbreitert und vertieft, wodurch im Hinblick auf mögliche Berufsfelder umfangreiche praktische Erfahrungen in verschiedenen Projekten und Gruppenzusammensetzungen erworben werden.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorprojekt aus einem der Fachgebiete der Angewandten Informatik zu wählen. Die wählbaren Projekte sind in UnivIS über das Schlagwort "AI-Projekt" auffindbar und als Projekte für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorprojekt 3</b> <b>Lehrformen:</b> Projektseminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS	<b>4,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben	
<b>Inhalte:</b>	

Die Inhalte der Bachelorprojekte werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.

**Literatur:**

Die Literatur wird zu Beginn eines Projekts von jedem anbietenden Fachgebiet bekannt gegeben.

**Prüfung**

Hausarbeit mit Kolloquium

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen.

Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden von der Betreuerin bzw. dem Betreuer des Projekts zu Beginn der Lehrveranstaltung bekannt gegeben.

<b>Modul AI-Seminar1-B Bachelorseminar 1 der Fächergruppe Angewandte Informatik</b> <i>Bachelor Seminar 1 in Applied Computer Science</i>		3 ECTS / 90 h 23 h Präsenzzeit 68 h Selbststudium
(seit SS25) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Angewandte Informatik)		
<b>Inhalte:</b> Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet der Angewandten Informatik mit wissenschaftlichen Methoden.		
<b>Lernziele/Kompetenzen:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik zu wählen. Die wählbaren Seminare sind in UnivIS über das Schlagwort "AI-Seminar" auffindbar und als Seminare für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorseminar 1</b> <b>Lehrformen:</b> Seminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS	<b>2,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben	
<b>Inhalte:</b> Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
<b>Literatur:</b> Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.	

<b>Prüfung</b>	
----------------	--

Hausarbeit mit Referat

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen.

Alternativ kann die Prüfungsleistung auf Hausarbeit mit Kolloquium festgelegt werden. 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.

<b>Modul AI-Seminar2-B Bachelorseminar 2 der Fächergruppe Angewandte Informatik</b> <i>Bachelor Seminar 2 in Applied Computer Science</i>		3 ECTS / 90 h 23 h Präsenzzeit 68 h Selbststudium
(seit SS25) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Angewandte Informatik)		
<b>Inhalte:</b> Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet der Angewandten Informatik mit wissenschaftlichen Methoden.		
<b>Lernziele/Kompetenzen:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.  In Relation zum Bachelorseminar 1 der Fächergruppe Angewandte Informatik werden dabei die Kompetenzen durch die Betrachtung eines weiteren Themas verbreitert und vertieft.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik Informatik zu wählen. Die wählbaren Seminare sind in UnivIS über das Schlagwort "AI-Seminar" auffindbar und als Seminare für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorseminar 2</b> <b>Lehrformen:</b> Seminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS	<b>2,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben	
<b>Inhalte:</b> Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
<b>Literatur:</b> Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.	

**Prüfung**

Hausarbeit mit Referat

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen.

Alternativ kann die Prüfungsleistung auf Hausarbeit mit Kolloquium festgelegt werden. 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.

<b>Modul AI-Thesis-B Bachelorarbeit im Studiengang Angewandte Informatik</b> <i>Bachelor Thesis in Applied Computer Science</i>		12 ECTS / 360 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: Professorinnen und Professoren der Angewandten Informatik		
<b>Inhalte:</b> 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 der Prüfungsordnung genannten Fächer zu entnehmen. Auf Antrag der Prüfungskandidatin bzw. des Prüfungskandidaten kann vom Prüfungsausschuss auch ein Thema aus einem anderen Fach zugelassen werden. In diesem Fall ist glaubhaft nachzuweisen, dass das gestellte Thema einen inhaltlichen Bezug zu dem zugrundeliegenden Studiengang aufweist.		
<b>Lernziele/Kompetenzen:</b> 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.  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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Die Zulassung setzt voraus, dass Module im Umfang von mindestens 120 ECTS-Punkten erfolgreich absolviert wurden.		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> 6.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Prüfung</b> Hausarbeit mit Kolloquium / Bearbeitungsfrist: 4 Monate	
---	--

<b>Modul AI-WebT-B Web-Technologien</b> <i>Web Technologies</i>		6 ECTS / 180 h
(seit SS26) Modulverantwortliche/r: Prof. Dr. Fabian Beck		
<b>Inhalte:</b> 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 betrachtet. Auch ausgewählte Qualitätsaspekte (z.B. Nutzerfreundlichkeit, Zugänglichkeit, Sicherheit, Performanz) von Web-Anwendungen werden thematisiert, sowie organisatorische und technische Entwicklungskonzepte.		
<b>Lernziele/Kompetenzen:</b> 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 selbstständig mit gängigen Frameworks und Techniken zu entwickeln und hierzu passende Technologie-Stacks auszuwählen. Sie sind in der Lage, Applikationen hinsichtlich verschiedener Qualitätsaspekte zu analysieren und die Entwicklung solcher Anwendungen effizient zu organisieren. Zudem können sie neue Trends und aufkommende Technologien im Bereich der Web-Entwicklung einschätzen und bewerten.		
<b>Sonstige Informationen:</b> Die Vorlesungen werden in der Regel in deutscher Sprache durchgeführt. Zahlreiche Quellen und Dokumentationen der Systeme sind aber auf Englisch. Übungen oder Aufgaben können ebenfalls in englischer Sprache erfolgen. Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Präsenzzeit in Vorlesung und Übung: 45h</li> <li>• Vor- und Nachbereitung der Vorlesung: 30h</li> <li>• Bearbeitung von Übungen und Studienleistungen: 75h</li> <li>• Vorbereitung zur Prüfung: 30h</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse der Informatik (z.B. Inf-Einf-B), insbesondere Programmierkenntnisse, werden erwartet.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Web-Technologien</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Fabian Beck <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich <hr/> <b>Lernziele:</b> siehe oben	<b>2,00 SWS</b>

<b>Inhalte:</b> siehe oben	
<b>Literatur:</b> wird in der Veranstaltung bekannt gegeben	
<b>2. Web-Technologien</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich <hr/> <b>Lernziele:</b> siehe oben <hr/> <b>Inhalte:</b> praktische Aufgaben zum Stoff der Vorlesung <hr/> <b>Literatur:</b> siehe Vorlesung	<b>2,00 SWS</b>

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> 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.	
---	--

<b>Modul AIC-IITP-B Internationales IT Projektmanagement</b> <i>International IT Project Management</i>		6 ECTS / 180 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Milad Mirbabaie		
<p><b>Inhalte:</b></p> <p>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.</p> <p>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.</p> <p>//</p> <p>The course teaches practical skills for managing and working on international IT projects. All phases of the project life cycle are covered: from initial analysis, planning and implementation through to monitoring and successful completion. Both traditional models such as the waterfall and V-model and agile methods such as Scrum are taken into account. Special attention is paid to the challenges of international teams and remote projects.</p>		
<p><b>Lernziele/Kompetenzen:</b></p> <p>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.</p> <p>//</p> <p>After successfully completing the course, students should be able to initiate, plan, manage and monitor IT projects in small and large organizations. They will be able to select suitable methods and process models from those covered, justify their choice, adapt the approaches to new and previously unknown problems and develop them further in a well-founded and appropriate manner based on the theoretical foundations.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b> Keine		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Internationales IT Projektmanagement</b> <b>Lehrformen:</b> Vorlesung und Übung		<b>4,00 SWS</b>

**Dozenten:** Prof. Dr. Milad Mirbabaie

**Sprache:** Englisch

**Angebotshäufigkeit:** SS, jährlich

**Lernziele:**

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.

//

After successfully completing the course, students should be able to initiate, plan, manage and monitor IT projects in small and large organizations. They will be able to select suitable methods and process models from those covered, justify their choice, adapt the approaches to new and previously unknown problems and develop them further in a well-founded and appropriate manner based on the theoretical foundations.

**Inhalte:**

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.

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.

//

The course teaches practical skills for managing and working on international IT projects. All phases of the project life cycle are covered: from initial analysis, planning and implementation through to monitoring and successful completion. Both traditional models such as the waterfall and V-model and agile methods such as Scrum are taken into account. Special attention is paid to the challenges of international teams and remote projects.

**Literatur:**

Wird im Kurs bekannt gegeben. // Will be announced in the course.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

<b>Modul AISE-DO-B DevOps für KI-Systeme</b> <i>DevOps für KI-Systeme</i>	6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Christoph Benzmüller	
<p><b>Inhalte:</b></p> <p>Dieser Kurs vermittelt den Teilnehmenden fundierte Kenntnisse und Fähigkeiten in der Implementierung, Verwaltung und Skalierung von KI-Systemen unter Verwendung modernster DevOps-Praktiken und Cloud-Computing-Infrastruktur. Die Inhalte konzentrieren sich auf:</p> <p><b>**Grundlagen der KI-Systeme und DevOps:**</b></p> <ul style="list-style-type: none"> <li>• Einführung in Künstliche Intelligenz, Maschinelles Lernen und Wissensbasierte Systeme</li> <li>• Überblick über DevOps-Prinzipien und -Praktiken</li> </ul> <p><b>**Cloud Computing:**</b></p> <ul style="list-style-type: none"> <li>• Einführung in Cloud-Service-Modelle (IaaS, PaaS, SaaS)</li> <li>• Benutzung, Bereitstellung und Verwaltung von KI-Systemen in der Cloud</li> <li>• Hardware-Beschleunigung für KI (GPUs, FPGAs, custom AI-chips, etc.)</li> </ul> <p><b>**Containerisierung und Orchestrierung:**</b></p> <ul style="list-style-type: none"> <li>• Erstellung und Verwaltung von Containern und Images. Bereitstellung in der Cloud.</li> <li>• Kubernetes-Grundlagen, Cluster-Erstellung, -Verwaltung und -Skalierung</li> </ul> <p><b>**Automatisierung und Continuous Integration/Continuous Deployment (CI/CD):**</b></p> <ul style="list-style-type: none"> <li>• Tools und Techniken für Automatisierung</li> <li>• Implementierung von CI/CD-Pipelines für KI-Anwendungen</li> </ul> <p><b>**Sicherheit, Datenschutz und ethische Aspekte:**</b></p> <ul style="list-style-type: none"> <li>• Überwachung (Monitoring) von KI-Systemen in der Cloud</li> <li>• Best Practices und Protokolle für Sicherheit und Datenschutz</li> <li>• Ethische Aspekte</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Am Ende des Kurses sollen die Teilnehmer in der Lage sein:</p> <ul style="list-style-type: none"> <li>• KI-Systeme effizient über den gesamten Entwicklungslebenszyklus zu verwalten und zu überwachen</li> <li>• Container und Kubernetes zur Skalierung und Verwaltung von KI-Anwendungen zu nutzen</li> <li>• CI/CD-Pipelines für eine schnelle und effiziente Entwicklung und Bereitstellung von KI-Anwendungen zu implementieren</li> <li>• Datenschutz- und Sicherheitsprotokolle für KI-Systeme in der Cloud zu verstehen und anzuwenden</li> <li>• Probleme und Herausforderungen bei der Nutzung und Implementierung von KI-Systemen in der Cloud zu identifizieren und Lösungen zu entwickeln</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p> <ul style="list-style-type: none"> <li>• Erfahrung mit einer oder mehreren Programmiersprachen (z.B. Python, Go, Java)</li> <li>• Grundlegende Kenntnisse über Maschinelles Lernen und KI</li> </ul>	<p><b>Besondere Bestehensvoraussetzungen:</b></p> <p>keine</p>

• Verständnis von Linux/Unix-Betriebssystemen und Shell-Scripting		
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. DevOps für KI-Systeme</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>2. DevOps für KI-Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>

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

<p><b>Modul AISE-LKR-B Logische Wissensrepräsentation und Schließen</b>  <i>Logical Knowledge Representation and Reasoning</i></p>	<p>6 ECTS / 180 h</p>
<p>(seit WS25/26)                  Modulverantwortliche/r: Prof. Dr. Christoph Benzmüller</p>	
<p><b>Inhalte:</b>                  Dieser Kurs bietet eine Einführung in die symbolische Wissensrepräsentation und das symbolische Schließen in der Künstlichen Intelligenz. Es werden theoretische Grundlagen, Methoden und Anwendungen der symbolischen KI besprochen, wobei insbesondere die Rolle von Logik und formaler Methoden bei der Repräsentation und Manipulation von Wissen betont wird. Als Alleinstellungsmerkmal wird dieser Kurs auch eine kurze Einführung in LogiKEy enthalten, eine logisch-pluralistische Wissensrepräsentations- und Schlussfolgerungsmethodik, die in der AISE-Gruppe aktuell entwickelt und angewendet wird.</p> <p>In den Kursinstanzen werden ausgewählte Aspekte aus der folgenden Themenliste behandelt:</p> <ul style="list-style-type: none"> <li>• Einführung in die Symbolische KI (z.B.: KI-Paradigmen, historische Entwicklung, Bedeutung der Symbolischen KI)</li> <li>• Grundlagen der Symbolischen Wissensrepräsentation (z.B.: klassische und nicht-klassische Logiken, Syntax und Semantik formaler Sprachen, Ontologien und Taxonomien, Beschreibungslogiken)</li> <li>• Inferenztechniken (z.B.: deduktives, induktives, abduktives, nicht-monotones Schließen)</li> <li>• Wissensrepräsentationstechniken (z.B.: semantische Netze, regelbasierte Systeme, Produktionssysteme, Frames und Skripte)</li> <li>• Fortgeschrittene Themen der Wissensrepräsentation (z.B.: normatives Schließen, zeitliches und räumliches Schließen, probabilistisches Schließen und Bayes'sche Netze, Multiagentensysteme, verteiltes und gemeinsames Wissen)</li> <li>• Anwendungen der Symbolischen KI (z.B.: automatisches und interaktives Theorembeweisen, Expertensysteme, Verarbeitung natürlicher Sprache, Planung und Terminierung, wissensbasierte Systeme in Medizin, Recht und Technik)</li> <li>• Integration mit Subsymbolischen KI-Ansätzen (z.B.: hybride Systeme, die symbolische und neuronale Ansätze kombinieren, semantische Netze und Wissensgraphen, aktuelle Trends und zukünftige Richtungen in der KI)</li> <li>• Fallstudien und praktische Implementierungen (z.B.: praktische Projekte und Aufgabenstellungen)</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <ul style="list-style-type: none"> <li>• Verständnis erwerben zu grundlegenden Konzepten und Techniken der symbolischen, insbesondere logik-basierten Wissensrepräsentation.</li> <li>• Kennenlernen verschiedener Methoden des symbolischen und logischen Schließens, die in der symbolischen KI verwendet werden.</li> <li>• Anwendungskompetenzen erwerben zum Einsatz symbolischen Schließens zur Lösung praktischer Probleme.</li> <li>• Verständnis erwerben zur Idee des universellen logischen Schließens.</li> <li>• Kompetenzen aufbauen zur Integration symbolischen und subsymbolischen Ansätzen in modernen KI-Systemen.</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>                  keine</p>	

<b>Empfohlene Vorkenntnisse:</b> Es wird empfohlen diese Veranstaltung erst ab dem 3. Semester zu belegen, nach Besuch weiterer Einführungsveranstaltungen in den Modulen A1, A2 und A3.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Logische Wissensrepräsentation und Schließen</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>2. Logische Wissensrepräsentation und Schließen</b> <b>Lehrformen:</b> Vorlesung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>

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

<b>Modul AlgoK-AK-B Algorithmen und Komplexität</b> <i>algorithms and complexity</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Isolde Adler		
<p><b>Inhalte:</b> Algorithms and problem solving lie at the heart of computer science.</p> <p>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?</p>		
<p><b>Lernziele/Kompetenzen:</b> Demonstrate an understanding of what constitutes an efficient and an inefficient solution to a computational problem,</p> <ul style="list-style-type: none"> <li>- Analyse the efficiency of algorithms,</li> <li>- Evaluate and justify appropriate ways to provide efficient solutions for computational problems,</li> <li>- Identify and apply different design principles in the design of algorithms,</li> <li>- Describe efficient algorithms for a range of computational problems, along with their computational complexity,</li> <li>- Describe the use of complexity classes and the relations between them,</li> <li>- Articulate the key concepts and critically evaluate approaches in a clear and rigorous manner,</li> <li>- Appreciate and understand in-depth the role of proofs in the area of algorithm design,</li> <li>- Recognise how the methods learned can be extended and used to solve other problems.</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b> Algorithms and data structures, basic knowledge of computability theory, proof techniques. Good English language skills.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>

1 Semester

**Lehrveranstaltungen****AlgoK-AK-B (Algorithmen und Komplexität)****2,00 SWS****Lehrformen:** Vorlesung und Übung**Sprache:** Englisch/Deutsch**Angebotshäufigkeit:** SS, jährlich**Lernziele:**

Demonstrate an understanding of what constitutes an efficient and an inefficient solution to a computational problem,

- Analyse the efficiency of algorithms,
- Evaluate and justify appropriate ways to provide efficient solutions for computational problems,
- Identify and apply different design principles in the design of algorithms,
- Describe efficient algorithms for a range of computational problems, along with their computational complexity,
- Describe the use of complexity classes and the relations between them,
- Articulate the key concepts and critically evaluate approaches in a clear and rigorous manner,
- 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.

**Inhalte:**

Algorithms and problem solving lie at the heart of computer science.

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**

mündliche Prüfung

**Beschreibung:**

Die Prüfungsdauer wird in der ersten LV bekannt gegeben

<b>Modul AlgoK-ALDAI-B Algorithms and logic in data science and AI</b>		6 ECTS / 180 h
<i>Algorithms and logic in data science and AI</i>		
(seit WS25/26)		
Modulverantwortliche/r: Prof. Dr. Isolde Adler		
Weitere Verantwortliche: Stefanie Dettmer		
<b>Inhalte:</b>		
<p>The past few decades saw an unprecedented increase in the size and complexity of data. We are surrounded by data in everyday life wherever we look.</p> <p>Understanding and analysing this data poses a huge challenge. Many current methods rely in neural networks and heuristics, at the price of sacrificing any guarantees on the computed output.</p> <p>However, guarantees are indispensable in many application areas, such</p> <p>In this module we discuss modern algorithmic approaches that come with guarantees both on the quality of the computed output and on the resource consumption.</p> <p>We also show lower bounds, i.e. infeasibility results. Furthermore, we uncover and exploit surprising connections between different areas of computer science.as aviation, cyber-security, finance, medicine and scientific research.</p>		
<b>Lernziele/Kompetenzen:</b>		
<p>In this module we discuss modern algorithmic approaches that come with guarantees both on the quality of the computed output and on the resource consumption.</p> <p>We also show lower bounds, i.e. infeasibility results. Furthermore, we uncover and exploit surprising connections between different areas of computer science.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
Successful completion of Inf-DM-B, Inf-LBR-B and AI-AuD-B. Solid mathematical skills, including proof techniques and first-order logic, and a general interest in theoretical methods		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere</b>
Preferable but not compulsory: successful completion of AlgoK-AK-B.		<b>Bestehensvoraussetzungen:</b>
		keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		Semester
<b>Lehrveranstaltungen</b>		
<b>Algorithms and logic in data science and AI</b>		<b>2,00 SWS</b>
<b>Lehrformen:</b> Vorlesung und Übung		
<b>Sprache:</b> Englisch		
<b>Angebotshäufigkeit:</b> WS, jährlich		
<b>Lernziele:</b>		
The topics include machine learning with guarantees, highly efficient property testing algorithms for huge graphs, networks and databases,		

exploiting tame structure of inputs via treewidth, designing fixed-parameter algorithms for notoriously hard problems, tackling various computational problems simultaneously through logic, highlighting applications in computational biology and exploiting connections between database query evaluation and constraint satisfaction in AI.

**Inhalte:**

The past few decades saw an unprecedented increase in the size and complexity of data. We are surrounded by data in everyday life wherever we look.

Understanding and analysing this data poses a huge challenge. Many current methods rely in neural networks and heuristics, at the price of sacrificing any guarantees on the computed output.

However, guarantees are indispensable in many application areas, such as In this module we discuss modern algorithmic approaches that come with guarantees both on the quality of the computed output and on the resource consumption.

We also show lower bounds, i.e. infeasibility results. Furthermore, we uncover and exploit surprising connections between different areas of computer science. aviation, cyber-security, finance, medicine and scientific research.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

Hinweis: Bei geringer Teilnehmerzahl evtl. mündliche Prüfung

<b>Modul CG-CGA-B Computergrafik und Animation</b> <i>Computer Graphics and Animation</i>	6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Sophie Jörg	
<p><b>Inhalte:</b></p> <p>Computergenerierte Inhalte sind weit verbreitet, z. B. in Filmen, virtuellen Welten oder Lernanwendungen. In dieser Lehrveranstaltung behandeln wir grundlegende Techniken und Algorithmen der dreidimensionalen Computergrafik und Animation. Themen beinhalten mathematische Grundlagen, Modellierung von dreidimensionalen Objekten, Raytracing, Reflexionsmodelle und Beleuchtung, Texturen, die Grafik-Pipeline, Grundlagen der Animation, Kinematik und Charakteranimation.</p> <p>Computer generated content is very common, for example, in movies, virtual worlds or educational applications. This course introduces students to the foundations of 3D computer graphics and animation. It provides an overview of different algorithms, and techniques in these fields. Topics include mathematical foundations, modelling of 3D objects, raytracing, shading and lighting, texturing, the graphics pipeline, introduction to animation, kinematics, and character animation.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Studierende sollen grundlegende Konzepte der Computergrafik und Animation definieren und erklären können. Sie sollen die mathematischen Grundlagen und gängige Methoden hierzu beherrschen.</p> <p>The goals include to be able to define and explain standard concepts in 3D computer graphics and animation. Students will learn the mathematical foundations and common methods.</p>	
<p><b>Sonstige Informationen:</b></p> <p>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> <li>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)</li> <li>• Semesterbegleitende Übungen: 22,5 Stunden (entspricht den 2 SWS Übungen)</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen: ca. 110 Stunden</li> <li>• Prüfungsvorbereitung und Prüfung: ca. 25 Stunden</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p> <p>Kenntnisse in linearer Algebra          Kenntnisse in der Programmierung (z.B. C++ oder Java)          Grundkenntnisse in der Medieninformatik</p> <p>z.B. können die Vorkenntnisse in folgenden Lehrveranstaltung erworben werden:</p> <p>Modul Algorithmen und Datenstrukturen (AI-AuD-B) - empfohlen          Modul Mathematik für Informatik 2 (Lineare Algebra) (KTR-MfI-2) - empfohlen, ebenso WiMa-B-001          Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen</p> <p>Knowledge in linear algebra.          Knowledge in programming (e.g., C++ or Java)          Basic knowledge in media informatics</p>	<p><b>Besondere Bestehensvoraussetzungen:</b></p> <p>keine</p>

<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
---	----------------------------------	---

<b>Lehrveranstaltungen</b>	
<b>1. Computergrafik und Animation</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Sophie Jörg <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>2. Computergrafik und Animation</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich <hr/> <b>Inhalte:</b> Übungen zum Vorlesungsstoff einschließlich der Berechnung und Programmierung von Beispielen.	<b>2,00 SWS</b>

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

<b>Modul COMNET-RN-B Rechnernetze</b> <i>Computer Networks</i>	6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: N.N. Weitere Verantwortliche: Prof. Dr. Florian Klingler	
<b>Inhalte:</b> Die Vorlesung Rechnernetze behandelt konzeptionelle und technologische Grundlagen von Rechnernetzen/Internet; thematisch werden dabei die Ebenen 1-4 des ISO/OSI-Modells abgedeckt. Zusätzlich werden Ansätze und Werkzeuge zur quantitativen Untersuchung von Kommunikationsprotokollen behandelt. Die Vorlesung wird durch eine Tafelübung begleitet. <ul style="list-style-type: none"> <li>• Physikalische Schicht: Signalausbreitung, Modulation, Shannon-Grenzen</li> <li>• Sicherungsschicht: ARQ, FEC, Framing. Medienzugriffsverfahren (Aloha, CSMA, CSMA/CD).</li> <li>• Netzwerkschicht: Routing als Graphproblem und als Netzproblem; Standardverfahren (Dijkstra, Bellmann-Ford); Routing vs. Forwarding; Fallstudie IP (longest prefix matching, BGP,...)</li> <li>• Transportschicht: Überlastabwehr, Flusskontrolle, Fairness, Fallstudie TCP.</li> <li>• Beschreibung von Diensten und Protokollen; quantitative Analyse von Kommunikationsprotokollen (z.B. Aloha, Markov-Kette für CSMA, Durchsatz bei TCP).</li> </ul>	
<b>Lernziele/Kompetenzen:</b> Absolventen der Lehrveranstaltung <ul style="list-style-type: none"> <li>• können die wesentlichen Aufgaben bei Konstruktion und Bau eines Rechnernetzes benennen und wesentliche Architekturansätze beschreiben;</li> <li>• können unterschiedliche Lösungen für ein Problem aufzählen, deren Vor- und Nachteile herausfinden und sich, gemäß der Anforderungen, für eine Lösung entscheiden;</li> <li>• Schwachstellen existierender Lösungen identifizieren und neue Kommunikationsprotokolle entwickeln und deren Leistungsfähigkeit bewerten.</li> </ul> Nichtkognitive Kompetenzen: <ul style="list-style-type: none"> <li>• Einsatz und Engagement</li> <li>• Lernkompetenz</li> </ul>	
<b>Sonstige Informationen:</b> Kontaktzeit: 60h Selbststudium: 120h	
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine	
<b>Empfohlene Vorkenntnisse:</b> <ul style="list-style-type: none"> <li>• erfolgreich abgeschlossene Prüfungen der Grundlagenfächer des Bachelorstudiums, insbesondere Einführung in Algorithmen, Programmierung und Software sowie grundlegende Kenntnisse effizienter Algorithmen</li> <li>• gute Programmierkenntnisse in C/C++ oder Python/etc.</li> <li>• Kenntnisse in Programmierung, Programmiersprachen, Software Engineering, Datenbanksysteme, Modellierung, Datenstrukturen und Algorithmen, Digitaltechnik, Analysis für Informatiker und Lineare Algebra für Informatiker sowie Systemsoftware</li> </ul>	<b>Besondere Bestehensvoraussetzungen:</b> keine

und systemnahe Programmierung oder vergleichbar werden empfohlen • für Studierende der alten Ordnungen (Studienbeginn vor WS 2024/25) sind statt der Module Inf-Einf-B und Inf-LBRR-B die Module DSG-EiAPS-B und Gdl-Mfl-1 empfohlene Vorkenntnisse  Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen Modul Logik und Berechenbarkeit (Inf-LBR-B) - empfohlen		
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Rechnernetze</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich <hr/> <b>Lernziele:</b> siehe Modulbeschreibung <hr/> <b>Inhalte:</b> siehe Modulbeschreibung	<b>4,00 SWS</b>

<b>Prüfung</b> Sonstiges <b>Beschreibung:</b> Die Prüfung wird als schriftliche Prüfung (120 Min) oder mündliche Prüfung (30 Min) durchgeführt.  Vom jeweiligen Lehrenden werden Art und Dauer der Prüfungsleistung spätestens in den ersten drei Wochen der Vorlesungszeit bekannt gegeben.	
---	--

<b>Modul DS-IDS-B Einführung in die Dialogsysteme</b> <i>Introduction to Dialogue Systems</i>		6 ECTS / 180 h
Modulverantwortliche/r: Prof. Dr. Stefan Ultes		
<b>Inhalte:</b> Dieses Modul befasst sich mit Dialog als sprachlichem Verhalten und seiner Modellierung in technischen Systemen. Es führt in das Gebiet der Sprachdialogtechnologie ein und beinhaltet die gesamte Verarbeitungskette eines Dialogsystems: akustische Signalverarbeitung, Spracherkennung, natürliches Sprachverstehen, Dialogmanagement, Sprachgenerierung und Sprachsynthese. Industrieunternehmen, die im Bereich der Sprachdialogsysteme arbeiten, werden an einzelnen Terminen Gastvorlesungen halten.  In der Übung werden ausgewählte Methoden und Sachverhalte vertieft und praktisch umgesetzt.		
<b>Lernziele/Kompetenzen:</b> Nach erfolgreicher Teilnahme an dem Kurs sollten Sie folgende Kenntnisse erworben haben: <ul style="list-style-type: none"> <li>• Allgemeines theoretisches Verständnis der Sprachdialogtechnologie</li> <li>• Verständnis von Dialogmodellierung und der üblichen Modularisierung dieser Aufgabe</li> <li>• Überblick über den aktuellen Stand der Technik für die sprachtechnologische Anwendung Dialogsystem</li> <li>• Kenntnis der Grundlagen der einzelnen Themengebiete eines modularen Dialogsystems</li> </ul>		
<b>Sonstige Informationen:</b> Die Arbeitsumfänge gestalten sich typischerweise wie folgt: <ul style="list-style-type: none"> <li>• Vorlesung: ~30h</li> <li>• Vor- und Nachbereitung der Vorlesung: ~30h</li> <li>• Prüfungsvorbereitung: ~30h</li> <li>• Übung: ~90h</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Module: Einführung in die KI [AI-KI-B]		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> Semester

<b>Lehrveranstaltungen</b>	
<b>1. Einführung in die Dialogsysteme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Stefan Ultes <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> siehe Modulbeschreibung	
<b>Inhalte:</b> In der Vorlesung werde unter anderem die folgenden Themen behandelt:	

<ul style="list-style-type: none"> <li>• Akustische Vorverarbeitung</li> <li>• Spracherkennung (Automatic Speech Recognition)</li> <li>• Sprachverstehen (Natural Language Understanding)</li> <li>• Dialogmanagement</li> <li>• Sprachgenerierung (Natural Language Generation)</li> <li>• Sprachsynthese</li> <li>• Statistische Sprachdialogsysteme</li> </ul> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Stuart Russell, Peter Norvig: "Artificial Intelligence: A Modern Approach"</li> <li>• Michael McTear: "Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots"</li> <li>• Dan Jurafsky, James H. Martin: "Speech and Language Processing"</li> <li>• Michael McTear: "Spoken Dialogue Systems Technology"</li> </ul>	
<p><b>2. Einführung in die Dialogsysteme</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Stefan Ultes</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b> siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b> In der Übung werden ausgewählte Methoden und Sachverhalte vertieft und parktisch umgesetzt.</p>	<p><b>2,00 SWS</b></p>

<p><b>Prüfung</b></p> <p>Sonstiges / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b></p> <p>Die Prüfungsform (mündlich oder schriftlich) wird zu Beginn des Semesters bekanntgegeben. Gegenstand der Prüfung sind sowohl alle Inhalte der Vorlesung (inklusive der Gastvorlesungen) als auch der Übung.</p>	
---	--

<b>Modul DT-CPP-B Einführung in die Systemprogrammierung in C++</b> <i>Introduction into Systems Programming in C++</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Maximilian Schüle		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> Systemprogrammierung in C++		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Einführung in die Systemprogrammierung in C++</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Maximilian Schüle <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>4,00 SWS</b>
<b>Inhalte:</b> 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.	
<b>Prüfung</b> Portfolio / Bearbeitungsfrist: 4 Monate	

<p><b>Modul EESYS-SaD-B Statistik und Data Science</b> <i>Statistics and Data Science</i></p>	<p>9 ECTS / 270 h</p>
<p>(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Thorsten Staake</p>	
<p><b>Inhalte:</b> Der Kurs bietet eine Einführung in die beschreibende und schließende Statistik und vermittelt neben statistischen Grundlagen auch Kompetenzen zur Datenanalyse mit der Statistik-Software R. Das vermittelte Wissen und die Anwendungskompetenzen bilden die Grundlagen z.B. für empirisch-quantitative Analysen und Theorieentwicklung, für das Data-Mining und für die Entwicklung bzw. Bewertungen von maschinellen Lernverfahren und KI-Systemen. Zahlreiche Phänomene werden anhand realer Daten illustriert und die vermittelten Methoden auf Daten zu aktuellen Fragestellungen angewendet.</p>	
<p><b>Lernziele/Kompetenzen:</b> Nach erfolgreichem Abschluss des Kurses sollen Teilnehmerinnen und Teilnehmer in der Lage sein,</p> <ol style="list-style-type: none"> <li>1. den Begriff Wahrscheinlichkeit und seine mathematische Umsetzung an einem Beispiel zu beschreiben und zu erläutern,</li> <li>2. die Bedeutung des Gesetzes der großen Zahlen und des zentralen Grenzwertsatzes der Statistik zu schildern,</li> <li>3. Konzepte zur Konstruktion von Schätzern (Maximum Likelihood Schätzung, Konfidenzintervall) und Hypothesentests an einem Beispiel zu beschreiben und zu erläutern,</li> <li>4. angemessene Kennzahlen und Verfahren zur Charakterisierung von empirischen Daten auszuwählen und zu berechnen,</li> <li>5. den Umgang mit grundlegenden diskreten und stetigen Verteilungen zu beherrschen und sie in einfachen Kontexten umzusetzen,</li> <li>6. zur Analyse univariater Daten Parameterschätzungen und Hypothesentests anzuwenden und deren Ergebnisse zu interpretieren,</li> <li>7. das Konzept der statistischen Signifikanz sowie deren Implikationen an einem Beispiel zu beschreiben und zu erläutern,</li> <li>8. für wichtige Verfahren (t-Tests, Regressionen) Anforderungen an die Daten benennen und Datensätze hinsichtlich dieser Anforderungen zu untersuchen,</li> <li>9. die Beziehung zwischen einer oder mehreren unabhängigen Variablen und einer abhängigen Variablen in einfachen Kontexten mithilfe von linearen Regressionsanalysen zu modellieren, die Ergebnisse zu interpretieren und die Passung der Modelle zu bewerten und</li> <li>10. mithilfe einer Programmiersprache Daten aufzubereiten und zu analysieren, einfache Modelle zu definieren und deren Parameter zu schätzen und die Ergebnisse zu visualisieren.</li> </ol>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p> <ul style="list-style-type: none"> <li>- Gute Deutsch- und Englischkenntnisse</li> <li>- Statistik-, Kombinatorik- und Analysis-Kenntnisse auf Abiturniveau</li> <li>- Grundlegende Fertigkeiten in funktionaler Programmierung</li> </ul>	<p><b>Besondere Bestehensvoraussetzungen:</b> keine</p>

<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
---	----------------------------------	---

<b>Lehrveranstaltungen</b>	
<p><b>Statistik und Data Science</b>  <b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Thorsten Staake  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>  Wahrscheinlichkeitsrechnung und Zählen</p> <ul style="list-style-type: none"> <li>- Stichprobenräume</li> <li>- Naive Definition der Wahrscheinlichkeit</li> <li>- Nicht-naive Definition der Wahrscheinlichkeit</li> </ul> <p>Bedingte Wahrscheinlichkeiten</p> <ul style="list-style-type: none"> <li>- Definition</li> <li>- Bayes-Regel, Regel der Gesamtwahrscheinlichkeit</li> <li>- Unabhängigkeit von Ereignissen</li> </ul> <p>Zufallsvariablen und ihre Verteilung</p> <ul style="list-style-type: none"> <li>- Zufallsvariablen</li> <li>- Verteilung und Wahrscheinlichkeitsmassefunktion</li> <li>- Erwartungswert, Varianz, Median, Modus</li> <li>- Diskrete Verteilungen: Uniform, Bernoulli, Binomial</li> <li>- Kumulative Verteilungsfunktionen diskreter Verteilungen</li> <li>- Funktionen von Zufallsvariablen</li> <li>- Unabhängigkeit von Zufallsvariablen</li> <li>- Verteilung und Wahrscheinlichkeitsdichtefunktion</li> <li>- Kontinuierliche Verteilungen: Gleichförmig, Normal, Exponentiell, Poisson</li> <li>- Kumulative Verteilungsfunktionen kontinuierlicher Verteilungen</li> <li>- Rechnen mit Erwartungswert und Varianz</li> </ul> <p>Stichproben / Sampling</p> <ul style="list-style-type: none"> <li>- Stichproben</li> <li>- Punkt- und Intervallschätzer: Mittelwerte / Standardfehler, Konfidenzintervalle</li> <li>- Standardabweichung vs. Standardfehler</li> <li>- Gesetz der großen Zahlen / Grenzwertsatz</li> </ul> <p>Statistischer Test / Hypothesentests</p> <ul style="list-style-type: none"> <li>- Hypothesen</li> <li>- Chi-Quadrat und Student-t</li> <li>- Test-Statistik</li> <li>- Typ-I / Typ-II Fehler</li> <li>- t-Test</li> <li>- Einstichproben- bzw. Zweistichproben-t-Test auf einen Mittelwertunterschied</li> <li>- Effektgrößen</li> </ul>	<b>3,00 SWS</b>

<ul style="list-style-type: none"> <li>- Statistische Erklärungskraft</li> </ul> <p>Korrelation, Kausalität, Experimente</p> <ul style="list-style-type: none"> <li>- Kausalität vs. Korrelation</li> <li>- Experimentelles Design</li> </ul> <p>Untersuchung gängiger Annahmen</p> <ul style="list-style-type: none"> <li>- Normalität / Shapiro-Wilk-Test</li> <li>- Homogenität der Varianz / Levene-Test</li> <li>- Umgang mit Ausreißern</li> </ul> <p>Regression</p> <ul style="list-style-type: none"> <li>- Methode der kleinsten Quadrate</li> <li>- Anpassungsgüte: Summe der Quadrate, R und <math>R^2</math></li> <li>- Einfache Regression</li> <li>- Interpretation der einfachen Regression</li> <li>- Multiple Regression</li> <li>- Ausreißer und einflussreiche Fälle</li> <li>- Zeitreihendaten</li> <li>- Robuste Standardfehler</li> <li>- Ggf. Binäre logistische Regression</li> </ul>	
<p><b>Literatur:</b> Frei verfügbare Online-Quellen, werden in der Vorlesung bekanntgegeben.</p>	

<p><b>Lehrveranstaltungen</b></p>	
<p><b>Statistik und Data Science</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Thorsten Staake  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <ul style="list-style-type: none"> <li>- Vertiefung, Übung und Anwendung der Inhalte der Vorlesung.</li> <li>- Einführung in die Software R.</li> <li>- Einsatz der Software R zur Aufbereitung von Daten, zur Modellbildung, zur Datenanalyse und zur Unterstützung der Interpretation der Ergebnisse.</li> </ul> <hr/> <p><b>Literatur:</b> Siehe Vorlesung</p>	<p><b>3,00 SWS</b></p>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> 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 bekanntgegeben, ob Studienleistungen angeboten werden. Falls Studienleistungen angeboten</p>	

---

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 bekanntgegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.	
--	--

<b>Modul ESE-SEL-B Software Engineering Lab</b> <i>Software Engineering Lab</i>		6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Matthias Galster		
<b>Inhalte:</b> Students will gain experience in developing software applications using modern techniques. Students work individually and in small groups to develop an application and practice fundamental skills required to develop software systems using modern tools and practices.		
<b>Lernziele/Kompetenzen:</b> After completing this course, students will be able to: <ul style="list-style-type: none"> <li>• Design a non-trivial piece of software using good development principles and practices</li> <li>• Understand, plan and document activities of a software development project</li> <li>• Analyse and solve technology-based problems through self-directed learning</li> <li>• Apply methods and tools to work and function in a team setting</li> <li>• Communicate effectively and efficiently with peers and non-technical audiences</li> </ul>		
<b>Sonstige Informationen:</b> The main language of instruction is English.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> none		
<b>Empfohlene Vorkenntnisse:</b> The course assumes that students have basic knowledge of computer science and software engineering (e.g., analysis, design, implementation and programming, testing).		<b>Besondere Bestehensvoraussetzungen:</b> none
<b>Angebotshäufigkeit:</b>	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> Semester

<b>Lehrveranstaltungen</b>	
<b>Software Engineering Lab</b> <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>4,00 SWS</b>
<b>Lernziele:</b> See module description.	
<b>Inhalte:</b> See module description.	
<b>Literatur:</b> Literature will be announced in the course.	

<b>Prüfung</b> Hausarbeit mit Kolloquium <b>Beschreibung:</b>	
---	--

---

Students will work on a project. At the end of the course, students will deliver a report and a presentation.	
---	--

<b>Modul ESE-SRE-B Software Requirements Engineering</b> <i>Software Requirements Engineering</i>		6 ECTS / 180 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Matthias Galster		
<b>Inhalte:</b> Many software engineering problems are rooted in not understanding and properly managing the needs of customers and other stakeholders during the development and maintenance of software. The course discusses concepts for systematically identifying, defining, and managing software requirements. The course will also explore the synergistic relationship between software requirements and software architecture design in iterative and incremental software development.		
<b>Lernziele/Kompetenzen:</b> After completing this course, students will be able to: <ul style="list-style-type: none"> <li>• Recognize the importance of systematically understanding and managing software requirements</li> <li>• Apply systematic requirements engineering practices for eliciting, documenting, and analyzing software requirements</li> <li>• Understand the relationship between software requirements and architecture design decisions</li> <li>• Identify and interpret the current state-of-the-art and practices of requirements engineering, including background literature</li> </ul>		
<b>Sonstige Informationen:</b> The main language of instruction is English.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> none		
<b>Empfohlene Vorkenntnisse:</b> The course assumes that students have basic knowledge of computer science and software engineering (e.g., analysis, design, implementation and programming, testing).  Recommended preparation: (1) SWT-FSE-B Foundations of Software Engineering; (2) INF-EINF-B Einführung in die Informatik		<b>Besondere Bestehensvoraussetzungen:</b> none
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Lecture</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Matthias Galster <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>2,00 SWS</b>
<b>Lernziele:</b> see module description		
<b>Inhalte:</b> see module description		
<b>Literatur:</b>		

Literature will be announced in the lectures.	
<b>2. Lab</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Matthias Galster <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> See module description.	
<b>Inhalte:</b> Labs will cover material presented in the lectures. Furthermore, students may deliver short presentations.	
<b>Literatur:</b> Literature will be announced in the lectures.	

<b>Prüfung</b> Sonstiges <b>Beschreibung:</b> The mode (oral or written) and duration of the examination will be announced in the first lecture.  The exam covers all course contents (lectures, labs).  The exam is set in English, but answers may be provided in either English or German. To pass the exam, students must achieve at least 50% of marks awarded for the exam.	
--	--

<b>Modul GAMES-Java-B Objektorientierte Programmierung mit Java</b> <i>Object-Oriented Programming with Java</i>		3 ECTS / 90 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Benedikt Morschheuser		
<b>Inhalte:</b> Das Modul "Objektorientierte Programmierung mit Java" vermittelt grundlegende Konzepte der imperativen und objektorientierten Programmierung anhand der weitverbreiteten Programmiersprache Java. Ziel des Moduls ist es, Studierende in die Lage zu versetzen, einfache Softwarelösungen selbstständig zu entwickeln und zu testen. Dabei stehen insbesondere die Prinzipien der objektorientierten Programmierung (z.B. Klassen, Objekte, Vererbung, Polymorphie und Abstraktion) zur Manipulation von grundlegenden Datentypen und einfachen Datenstrukturen (z.B. Felder) im Mittelpunkt. Anhand von Beispielen und durch einfache Programmieraufgaben wird die praktische Anwendung des vermittelten Wissens trainiert. Die Prüfungsleistung besteht aus einer Klausur.		
<b>Lernziele/Kompetenzen:</b> Lernziele: <ul style="list-style-type: none"> <li>• Die Studierenden beherrschen grundlegende Konzepte der objektorientierten Programmierung und können diese gezielt zur Entwicklung einfacher Programme in Java anwenden.</li> <li>• Die Studierenden können eigenständig kleinere Programme in Java entwickeln und bestehende Programmfragmente systematisch auf Fehler analysieren.</li> <li>• Die Studierenden sind in der Lage, einfache Datenstrukturen wie Felder zu manipulieren und einfache algorithmische Probleme unter Einsatz geeigneter Kontrollstrukturen (z.B. Schleifen, Bedingungen) zu lösen.</li> <li>• Die Studierenden trainieren die Fähigkeit, Problemstellungen, wie sie in wirtschaftlichen Anwendungskontexten bestehen, mit algorithmischen Lösungsansätzen zu bearbeiten.</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Teilnahme an der Grundlagenvorlesung Inf-Einf-B, in der Kenntnisse zu den Grundprinzipien der Informatik, Algorithmen und Kontrollstrukturen vermittelt werden.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Objektorientierte Programmierung mit Java</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Benedikt Morschheuser <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> Lernziele:	

- Die Studierenden beherrschen grundlegende Konzepte der objektorientierten Programmierung und können diese gezielt zur Entwicklung einfacher Programme in Java anwenden.
- Die Studierenden sind in der Lage, einfache Datenstrukturen wie Felder zu manipulieren und einfache algorithmische Probleme unter Einsatz geeigneter Kontrollstrukturen (z.B. Schleifen, Bedingungen) zu lösen.
- Studierende lernen bestehende Programmfragmente systematisch auf Fehler zu analysieren und diese zu beheben.
- Die Studierenden trainieren die Fähigkeit, Problemstellungen, wie sie in wirtschaftlichen Anwendungskontexten bestehen, mit algorithmischen Lösungsansätzen zu bearbeiten.

**Inhalte:**

Objektorientierte Programmierung (OOP) ermöglicht die Modellierung komplexer Softwaresysteme als Zusammenspiel selbständiger, wiederverwendbarer Komponenten. In Industrie und Forschung findet OOP breite Anwendung – von Unternehmensanwendungen und Webdiensten über mobile Apps bis hin zu Simulationen und Spieleentwicklung. Objektorientierte Paradigmen fördern Wartbarkeit, Erweiterbarkeit und Teamarbeit und sind daher ein zentrales Element moderner Softwareentwicklung. Sie ebnen den Weg für die Realisierung skalierbarer und zukunftssicherer Anwendungen.

Aufbauend auf dem Konzept der objektorientierten Programmierung betrachtet diese Vorlesung zentrale Aspekte der Softwareentwicklung. Studierende lernen praktische Problemlösungen und Algorithmen am Beispiel der Programmiersprache Java kennen und entwickeln selbst eigene Programme.

Im Rahmen der Vorlesung werden nach einer Einführung in das Thema die folgenden Themen theoretische, methodisch und teilweise technisch behandelt:

- Prinzipien objektorientierter Programmierung
- Weiterführende Konzepte objektorientierter Programmierung (z.B. Polymorphie, Abstraktion)
- Manipulation grundlegender Datentypen und Datenstrukturen in Java
- Algorithmische Problemlösung mit Hilfe von Kontrollstrukturen in Java
- Aufbau und Strukturierung zusammenhängender objektorientierter Programm

**Literatur:**

Wird in der Vorlesung bekannt gegeben.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

<b>Modul Gdl-MTL-B Modal and Temporal Logic</b> <i>Modal and Temporal Logic</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> Nicht nur die Verifikation der funktionalen Korrektheit von Algorithmen und die funktionale Analyse verteilter und verlässlicher Systeme erfordert logisch-symbolische Verfahren. Auch viele Steuerungsprobleme in Anwendungsfeldern wie der Automatisierung von Wirtschaftsprozessen, intelligenten autonomen Agenten oder in Sicherheitsprotokollen lassen sich nur schwer mit herkömmlichen analytisch-numerischen Methoden behandeln. Dank der sich kontinuierlich verbessernden Leistungsfähigkeit moderner Rechner und der Erfolge im Gebiet der <i>Computational Logic</i> kommt der formalen Logik in der Informationstechnik wachsende Bedeutung zu. Die Vorlesung gibt einen Einblick in die Familie der Modallogiken als die wichtigsten informatikrelevanten Logiken, stellt zugehörige Implementierungstechniken und Entscheidungsverfahren vor und zeigt typische Anwendungen auf.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> grundlegende Programmierkenntnisse, gute Englischkenntnisse Modul Diskrete Modellierung (Inf-DM-B) - empfohlen Modul Logik und Berechenbarkeit (Inf-LBR-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Modal and Temporal Logic</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Ph.D. Michael Mendler <b>Sprache:</b> Englisch/Deutsch		<b>4,00 SWS</b>

**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 schriftliche Prüfung (90 Minuten) durchgeführt. Die Prüfungsform wird den Teilnehmern am Anfang des Semesters bekanntgegeben.

<b>Modul HCI-DISTP-B Design Interaktiver Systeme: Theorie und Praxis</b> <i>Design of Interactive Systems: Theory and Practice</i>		6 ECTS / 180 h
(seit SS24) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Das Modul vermittelt theoretische, methodische und praktische Grundlagen des Entwurfs sowie praktisches Entwerfen mit Fokus auf eine forschungsrelevante Aufgabenstellung. Es besteht aus einem Vorlesungs- und Übungsteil, in dem die Theorie und Praxis vermittelt wird. Der Reflexion über die Theorie und Praxis ist eine eigene Lehrveranstaltung gewidmet.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die allgemeine Einführung und Vermittlung grundlegender Praktiken, Prozesse und Methoden des Designs mit besonderem, anwendungsbezogenem Fokus auf die nutzerzentrierte Gestaltung komplexer, interaktiver Systeme sowie der Reflexion darüber		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a>  Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesungseinheiten</li> <li>• Teilnahme an Gruppenbesprechungen</li> <li>• Bearbeitung der Aufgabenstellung allein und im Team</li> <li>• Vorbereitung von Besprechungen und Präsentationen</li> <li>• Prüfungsvorbereitung</li> </ul> Die Aufwände können dabei in Abhängigkeit von der Aufgabenstellung und der in der Gruppe abgestimmten Aufgabenverteilung unter den Gruppenmitgliedern unterschiedlich auf die Bereiche verteilt sein.  Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Design Interaktiver Systeme: Theorie und Praxis</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Jochen Denzinger <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>1,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden folgende Themen behandelt:	

<ul style="list-style-type: none"> <li>• Designtheorie und -geschichte</li> <li>• Gestaltung multimodaler Benutzungsoberflächen</li> <li>• User-Centered Design, User Experience Design</li> <li>• Entwurfspraxis inkl. praktischer Einsatz von Methoden für den iterativen Entwurf</li> </ul> <p>In der Übung werden wechselnde Aufgaben zu den Inhalten der Lehrveranstaltung bearbeitet. Im Rahmen der Veranstaltung ist ein iterativer Entwurf als praktische Übung von den Studierenden zu erstellen. Die bearbeitete Aufgabenstellung geht deutlich über den Umfang einer normalen Übungsaufgabe hinaus und wird in kleinen Gruppen bearbeitet. Das erarbeitete Ergebnis wird dokumentiert und in einer Abschlusspräsentation vorgestellt.</p>	
<p><b>Literatur:</b></p> <p>Die Veranstaltung ist eine Zusammensetzung verschiedener Quellen; als ergänzende Quellen und zum Nachschlagen wir u.a. empfohlen:</p> <ul style="list-style-type: none"> <li>• Krippendorff, K. The Semantic Turn. A New Foundation for Design. Taylor &amp; Francis Group, Boca Raton, FL, 2006.</li> <li>• Moggridge, B. Designing Interactions. MIT Press, Cambridge, MA, 2007.</li> </ul>	
<p><b>Prüfung</b> Kolloquium / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b> Kolloquium zum Übungsverlauf und Übungsergebnissen</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>Reflexion zum Design interaktiver Systeme: Theorie und Praxis</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Jochen Denzinger</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>In der Übung erlernen die Studierenden die kritische Reflexion der eigenen Bearbeitung der wechselnden Aufgaben zu den Inhalten der Lehrveranstaltung im Rahmen des eigenen iterativen Entwurfs.</p> <hr/> <p><b>Literatur:</b></p> <p>Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen</p>	<p><b>1,00 SWS</b></p>
<p><b>Prüfung</b> Kolloquium / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b> Kolloquium zum Übungsverlauf und Übungsergebnissen</p>	

<b>Modul HCI-IS-B Interaktive Systeme</b> <i>Interactive Systems</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit SS23) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen der Mensch-Computer-Interaktion.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden</li> <li>• Bearbeiten der optionalen Studienleistungen: insgesamt ca. 45 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik im Umfang einer Einführung in die Informatik		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Interaktive Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden die folgenden Themen behandelt: <ul style="list-style-type: none"> <li>• Einführung in die Gestaltung von Benutzungsoberflächen</li> <li>• Benutzer und Humanfaktoren</li> <li>• Maschinen und technische Faktoren</li> <li>• Interaktion, Entwurf, Prototyping und Entwicklung</li> </ul>	

<ul style="list-style-type: none"> <li>• Evaluierung von interaktiven Systemen</li> <li>• Entwicklungsprozess interaktiver Systeme</li> <li>• Interaktive Systeme im größeren Kontext und verwandte Themen</li> </ul> <p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Sharp, H., Rogers, Y. and Preece, J. Interaction Design: Beyond Human-Computer Interaction. (5th ed.). Wiley, New York, 2019</li> <li>• Dix, A., Finlay, J., Abowd, G.D. und Beale, R. Human-Computer Interaction. Pearson, Englewood Cliffs, NJ, 3. Auflage, 2004.</li> </ul>	
<p><b>2. Interaktive Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	<b>2,00 SWS</b>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> 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.</p> <p>In der Klausur über 90 min. können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die 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.</p>	

<p><b>Prüfung</b> mündliche Prüfung</p> <p><b>Beschreibung:</b> 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.</p>	
---	--

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.

<b>Modul HCI-KS-B Kooperative Systeme</b> <i>Cooperative Systems</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen der rechnergestützten Gruppenarbeit.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die Vermittlung von grundlegenden Paradigmen und Konzepten von Rechnergestützter Gruppenarbeit (Computer-Supported Cooperative Work; CSCW) sowie die daraus resultierenden Designprinzipien und Prototypen. Dabei wird der Begriff breit gefasst; das zentrale Anliegen ist entsprechend die generelle technische Unterstützung von sozialer Interaktion, welche vom gemeinsamen Arbeiten und Lernen bis zum privaten Austausch reichen kann.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden</li> <li>• Bearbeiten der optionalen Studienleistungeng: insgesamt ca. 45 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik im Umfang einer Einführung in Algorithmen, Programmierung und Software, sowie Grundkenntnisse in Webtechnologien.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Kooperative Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden nach einer Einführung in das Thema die folgenden Themen behandelt: <ul style="list-style-type: none"> <li>• Grundlegende Konzepte</li> </ul>	

<ul style="list-style-type: none"> <li>• Technologische Unterstützung für wechselseitige Information, Kommunikation, Koordination, Gruppenarbeit und Online-Gemeinschaften</li> <li>• Analyse kooperativer Umgebungen</li> <li>• Entwurf von CSCW und Groupware</li> <li>• Implementation von CSCW und Groupware</li> <li>• CSCW im größeren Kontext und verwandte Themen</li> </ul>	
<p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Gross, T. und Koch, M. Computer-Supported Cooperative Work. Oldenbourg, München, 2007.</li> <li>• Borghoff, U.M. und Schlichter, J.H. Computer-Supported Cooperative Work: Introduction to Distributed Applications. Springer-Verlag, Heidelberg, 2000.</li> </ul>	
<p><b>Prüfung</b> mündliche Prüfung</p> <p><b>Beschreibung:</b> 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.</p> <p>In der mündlichen Prüfung können 90 Punkte erzielt werden. Die Prüfungsdauer wird im ersten Veranstaltungstermin mitgeteilt.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die 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.</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>Kooperative Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

**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 Klausur über 90 min. können 90 Punkte erzielt werden.

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.

<b>Modul HCI-US-B Ubiquitäre Systeme</b> <i>Ubiquitous Systems</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen des Ubiquitous Computing.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen sowie Bearbeitung der optionalen Studienleistungen): ca. 75 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Modul Algorithmen und Datenstrukturen (AI-AuD-B) Modul Einführung in die Informatik (Inf-Einf-B)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Ubiquitäre Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden nach einer Einführung in das Thema Ubiquitous Computing - also der allgegenwärtigen Rechner, die verschwindend klein, teilweise in Alltagsgegenständen eingebaut, als Client und Server fungieren und miteinander kommunizieren können - die folgenden Themen konzeptionell, technisch und methodisch behandelt:	

<ul style="list-style-type: none"> <li>• Grundlegende Konzepte</li> <li>• Basistechnologie und Infrastrukturen</li> <li>• Ubiquitäre Systeme und Prototypen</li> <li>• Kontextadaptivität</li> <li>• Benutzerinteraktion</li> <li>• Ubiquitäre Systeme im größeren Kontext und verwandte Themen</li> </ul>	
<p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Krumm, J., (Hrsg.). Ubiquitous Computing Fundamentals. Taylor &amp; Francis Group, Boca Raton, FL, 2010.</li> </ul>	
<p><b>Prüfung</b> mündliche Prüfung</p> <p><b>Beschreibung:</b> 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.</p> <p>In der mündlichen Prüfung können 90 Punkte erzielt werden. Die Prüfungsdauer wird im ersten Veranstaltungstermin mitgeteilt.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die 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.</p>	
<b>Lehrveranstaltungen</b>	
<p><b>Ubiquitäre Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen.</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	<b>2,00 SWS</b>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b></p>	

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 Klausur über 90 min. können 90 Punkte erzielt werden.

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.

<b>Modul Inf-DM-B Diskrete Modellierung</b> <i>Discrete Modelling</i>		9 ECTS / 270 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Isolde Adler		
<p><b>Inhalte:</b> 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. 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. 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.</p>		
<p><b>Lernziele/Kompetenzen:</b> Vertrautheit mit unterschiedlichen Modellierungsmethoden Sicherheit im mathematisch exakten Argumentieren Vertrautheit mit grundlegenden Definitionen und Eigenschaften aus dem Bereich der diskreten Mathematik und mit deren Rolle in der Informatik Sicherheit in der Entwicklung von Strategien zur Problemlösung Analytische Fähigkeiten</p>		
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine</p>		
<p><b>Empfohlene Vorkenntnisse:</b> Interesse an formalen Methoden. Dies ist eine grundlegende Veranstaltung, die für die ersten Studiensemester empfohlen wird.</p>		<p><b>Besondere Bestehensvoraussetzungen:</b> keine</p>
<p><b>Angebotshäufigkeit:</b> WS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b> 1 Semester</p>

<b>Lehrveranstaltungen</b>	
<p><b>Diskrete Modellierung</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Isolde Adler <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p>	<p><b>6,00 SWS</b></p>
<b>Inhalte:</b>	

---

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.	
---	--

<b>Prüfung</b>	
----------------	--

schriftliche Prüfung (Klausur)	
--------------------------------	--

<b>Beschreibung:</b>	
----------------------	--

Examination: written exam (e-exam), duration: 120 minutes.	
--	--

Schriftliche Prüfung (E-Prüfung), Prüfungsdauer: 120 Minuten.	
---	--

<b>Modul Inf-Einf-B Einführung in die Informatik</b> <i>Introduction to Computer Science</i>	9 ECTS / 270 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Dominik Herrmann	
<p><b>Inhalte:</b></p> <p>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.</p> <p>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.</p> <p>Das Modul bietet zudem einen ersten Einblick in Algorithmen (grundlegende Such- und Sortierverfahren) sowie abstrakte Datentypen und gängige Datenstrukturen (einfach verkettete Listen, Stacks, Tries). Die Studierenden implementieren iterative und rekursive Algorithmen zur Lösung grundlegender Probleme und modellieren Problemlösungen mit passenden Datenstrukturen wie Listen, Bäumen, Tries und Hash-Tabellen. Ergänzend erhalten sie Einblicke in die für die Anwendungsprogrammierung relevanten Grundlagen 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.</p> <p>Die Inhalte werden theoretisch fundiert; der Schwerpunkt des Moduls liegt jedoch auf der Entwicklung praktischer Problemlösungskompetenzen durch Übungsaufgaben, die ein intensives Selbststudium erfordern. Das Erlernete wird durch ein semesterbegleitendes Programmierprojekt, das am Ende des Semesters präsentiert wird, angewandt und gefestigt.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Lernziele auf den Kompetenzniveaus Wissen, Verstehen und Analysieren:</p> <ul style="list-style-type: none"> <li>• Die Studierenden verstehen grundlegende Konzepte der Programmierung in C und Python (etwa Datentypen, Variablen und Kontrollstrukturen) und können erklären, wie diese funktionieren.</li> <li>• Sie können die grundlegende Funktionsweise der Speicherverwaltung in C (z.B. Stack vs. Heap, Pointer-Arithmetik) und binäre und hexadezimale Zahlendarstellung erklären und Situationen analysieren, in denen diese Konzepte vorkommen.</li> <li>• Sie verstehen verschiedene Abstraktionsebenen eines Programms (etwa Funktionen und Bibliotheken) und können zwischen Design- und Implementierungsdetails unterscheiden.</li> <li>• Sie erkennen Problemstellungen oder Lösungen, bei denen grundlegende Paradigmen wie Abstraktion oder "Teile und Herrsche" eine Rolle spielen.</li> </ul>	

- Sie können Algorithmen und Datenstrukturen (z.B. lineare und binäre Suche, Selection Sort, Bubble Sort, Merge Sort, verkettete Listen, Tries) erklären und grundlegend hinsichtlich ihrer Eigenschaften (z.B. Laufzeit vs. Speicherplatz) analysieren.
- Sie verstehen die Konzepte der prozeduralen (etwa in C), objektorientierten und funktionalen Programmierung (z.B. in Python). Die können Probleme analysieren und erkennen, welche Konzepte für eine Lösung geeignet sind.
- Weiterhin können sie die für die Anwendungsprogrammierung relevanten Grundlagen von Rechnernetzen (TCP/IP) erklären und das Zusammenspiel der Komponenten einfacher Client-Server-Webanwendungen (Webseiten mit HTML, CSS, JavaScript, Python-Skripte auf dem Webserver) erklären.

Lernziele auf den Kompetenzniveaus Anwenden und Erschaffen:

- Die Studierenden können selbstständig Programme in C und Python entwickeln, die gegebene Problemstellungen lösen (Berechnungen, Generierung und Analyse von Zahlenfolgen, Verarbeitung von Texten und strukturierten Daten)
- Sie können Probleme in Teilprobleme zerlegen, mit selbst erstellten Funktionen und Datenstrukturen implementieren und dabei auf Funktionen aus grundlegenden Programmierbibliotheken (in C etwa `stdlib.h`, `stdio.h`, `string.h`) zurückgreifen.
- Sie beherrschen den praktischen Umgang mit Entwicklungswerkzeugen (etwa Compiler und Debugger) und Kommandozeile (Nutzung von Shell-Programmen wie `cp`, `mv`, `ls`, `cat` und `sort`)
- Sie können Programme systematisch testen (z.B. mit `pytest`) und Fehler identifizieren und beheben
- Sie können dynamische Datenstrukturen (etwa verkettete Listen und Stacks) zur Problemlösung verwenden diese unter Verwendung von dynamischer Speicherverwaltung erzeugen (in C mit `malloc` und `free`)
- Sie können Edge Cases erkennen (z.B. Eingabevalidierung, Grenzfälle bei Arrays) und ihre Programme entsprechend robust implementieren.
- Sie können vorhandenen Code hinsichtlich Korrektheit, Design und Stil beurteilen und verbessern.
- Sie sind in der Lage, sich selbstständig in neue Programmiersprachen und -konzepte einzuarbeiten.
- Sie können mit den erworbenen Kompetenzen eine selbstgewählte Aufgabenstellung analysieren, programmatisch lösen und ihre Lösung präsentieren.

Die hier aufgeführten Lernziele können an aktuelle Entwicklungen angepasst werden. Änderungen werden in der Vorlesung bekannt gegeben.

**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 diesen 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

<b>Empfohlene Vorkenntnisse:</b> Dieses Modul setzt keine Programmierkenntnisse voraus.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung</b>  <b>Lehrformen:</b> Vorlesung  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Literatur:</b>  Für diesen Kurs werden keine Bücher benötigt oder empfohlen. Die unten aufgeführten Bücher könnten jedoch von Interesse sein.</p> <ul style="list-style-type: none"> <li>• Hacker's Delight, Zweite Ausgabe, Henry S. Warren Jr., Pearson Education, 2013.</li> <li>• How Computers Work, Zehnte Ausgabe, Ron White, Que Publishing, 2014.</li> <li>• Programmieren in C, Vierte Ausgabe, Stephen G. Kochan, Pearson Education, 2015.</li> <li>• Think Like a Programmer: An Introduction to Creative Problem Solving, V. Anton Spraul, No Starch Press, 2012.</li> <li>• Automate the Boring Stuff with Python, 3rd Edition, Al Sweigart, No Starch Press, 2025.</li> <li>• Python Programming Exercises, Gently Explained, Al Sweigart, 2022.</li> </ul>	<b>4,00 SWS</b>
<p><b>2. Übung</b>  <b>Lehrformen:</b> Übung  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>  In der Übung werden die wichtigsten Konzepte der Vorlesung an praktischen Beispielen veranschaulicht und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema vertieft.</p>	<b>2,00 SWS</b>

<p><b>Prüfung</b>  schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 180 Minuten</p> <p><b>Beschreibung:</b>  Es können semesterbegleitende Studienleistungen erbracht werden. Informationen zu Anforderungen, Art und Umfang, Bearbeitungsfrist und der maximal erreichbaren Anzahl der dadurch erreichbaren Bonuspunkte 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 die Hälfte der maximal erreichbaren Punkte erreicht wird), werden die Bonuspunkte zu den in der Prüfung erreichten Punkten addiert. Die Note 1,0 kann auch ohne die Bonuspunkte erreicht werden.</p>	
--	--

<b>Modul Inf-GRABS-B Grundlagen der Rechnerarchitektur und Betriebssysteme</b> <i>Foundations of Computer Architecture and Operating Systems</i>		9 ECTS / 270 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Michael Engel		
<b>Inhalte:</b> 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. 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. 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. 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.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Programmierkenntnisse, z.B. aus EIAPS oder der Nachfolgeveranstaltung Inf-Einf-B, vergleichbare Programmierkenntnisse, z.B. aus dem Modul Inf-Prof-C-B, das parallel belegt werden kann.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>

	1 Semester
<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung Grundlagen der Rechnerarchitektur und Betriebssysteme</b></p> <p><b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Michael Engel  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>  siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b>  siehe Modulbeschreibung</p> <hr/> <p><b>Literatur:</b>  Frank Slomka, Michael Glaß  <b>Grundlagen der Rechnerarchitektur: Von der Schaltung zum Prozessor</b>  Springer Fachmedien Wiesbaden GmbH, 1. Auflage 2023  ISBN-13: 978-3-658-36658-2 (Softcover) / 978-3-658-36659-9 (eBook)</p> Andrew Waterman, David A. Patterson <b>The RISC-V Reader: An Open Architecture Atlas</b> Strawberry Canyon, 2017. ISBN-13: 978-0-999-24911-6 Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau <b>Operating Systems: Three Easy Pieces</b> <a href="https://pages.cs.wisc.edu/~remzi/OSTEP/">https://pages.cs.wisc.edu/~remzi/OSTEP/</a> Weitere Literatur wird nach Bedarf zur Verfügung gestellt.	4,00 SWS
<p><b>2. Übung Grundlagen der Rechnerarchitektur und Betriebssysteme</b></p> <p><b>Lehrformen:</b> Praktikum, Übung  <b>Dozenten:</b> Prof. Dr. Michael Engel  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>  siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b>  siehe Modulbeschreibung</p>	2,00 SWS
<p><b>Prüfung</b>  schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 90 Minuten  Bearbeitungsfrist: 90 Minuten</p>	

<b>Modul Inf-LBR-B Logik und Berechenbarkeit</b> <i>Logic and Computability</i>		9 ECTS / 270 h
(seit WS25/26) Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> 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.  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.		
<b>Sonstige Informationen:</b> 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.  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.  Der Arbeitsaufwand für dieses Modul gliedert sich insgesamt grob wie folgt:		
<ul style="list-style-type: none"> <li>• Teilnahme an Vorlesungen und Übungen: 67 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 65 Stunden</li> <li>• Prüfungsvorbereitung + Teilnahme and Prüfungen: 48 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> gute Englischkenntnisse Modul Diskrete Modellierung (Inf-DM-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>

1 Semester

**Lehrveranstaltungen****Logik und Berechenbarkeit****6,00 SWS****Lehrformen:** Vorlesung und Übung**Dozenten:** Prof. Ph.D. Michael Mendler, N.N.**Sprache:** Deutsch/Englisch**Angebotshäufigkeit:** SS, jährlich**Lernziele:**

Teil1:

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.

Teil 2:

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).

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

<b>Modul Inf-Ment-B Studieneinstiegsmentoring für Bachelorstudierende</b> <i>Introductory Mentoring for Bachelor Students</i>		1 ECTS / 30 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas Weitere Verantwortliche: Prof. Dr. Fabian Beck		
<b>Inhalte:</b> Die Veranstaltung bietet allgemeine Informationen zur Universität und Fakultät, und klärt Fragen der Studienorganisation. In von Tutor:innen betreuten Kleingruppen werden Lernstrategien und berufliche Entwicklungsmöglichkeiten detailliert erarbeitet und diskutiert.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden können eigenständig den Ablauf ihres Studiums organisieren sowie den Studienverlauf informiert planen. Durch erste Diskurserfahrungen und Impulse zur Meinungsbildung sind sie befähigt, an hochschulinternen Diskussionen zu partizipieren und Prozesse mitzugestalten. Sie entwickeln eigene, fachspezifische Lernstrategien, können diese reflektieren und in Lerngruppen einsetzen. Sie wissen bei Problemen um persönliche Handlungsmöglichkeiten und Unterstützungsangebote und können auf Erfahrungen von Mitstudierende zurückgreifen. Aufbauend auf ihr Wissen über berufliche und fachliche Perspektiven sowie die persönlichen Perspektiven anderer, können sie Wahl- und Zusatzangebote im Studium zielgerichtet wahrnehmen und ein eigenes berufliches Profil entwickeln. Sie sind geübt im persönlichen, studienbezogenen Austausch und können unterschiedliche Standpunkte im fachlichen Diskurs wertschätzen.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand gliedert sich in etwa wie folgt: <ul style="list-style-type: none"> <li>• 6 h Teilnahme an Vorlesungen</li> <li>• 12 h Teilnahme an Mentoring-Treffen in Kleingruppen</li> <li>• 12 h Vor- und Nachbereitung, Reflexion, Einzelberatung</li> </ul> Die kontinuierliche, vertrauliche und arbeitsteilige Erarbeitung der Lerninhalte in Kleingruppen erfordert eine regelmäßige Teilnahme an der Lehrveranstaltung (Anwesenheitspflicht).		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Benotung:</b> Das Modul ist unbenotet!		
<b>Lehrveranstaltungen</b>		
<b>1. Studieneinstiegsmentoring für Bachelorstudierende</b> <b>Lehrformen:</b> Vorlesung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS		<b>0,25 SWS</b>

<b>Inhalte:</b> <ul style="list-style-type: none"> <li>• Aufbau und Angebote der Universität und Fakultät</li> <li>• Studium und Studienorganisation</li> <li>• Effizientes Lernen und Prüfungsvorbereitung</li> <li>• Berufliche Karriere und Forschung im Bereich Informatik</li> </ul>	
<b>Literatur:</b> Webseiten der Universität und Fakultät	
<b>2. Studieneinstiegsmentoring für Bachelorstudierende</b> <b>Lehrformen:</b> Kleingruppenprojekt <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS	<b>0,50 SWS</b>
<b>Prüfung</b> Portfolio / Bearbeitungsfrist: 4 Monate, unbenotet <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Persönliche Reflexion der Lerninhalte	

<b>Modul Inf-Prog-C-B Einführung in die C-Programmierung</b> <i>Introduction to Programming in C</i>		3 ECTS / 90 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Michael Engel		
<b>Inhalte:</b> Die Vorlesung führt in die Programmiersprache C ein, aufbauend auf imperativen Sprachkonstrukten (Sequenz, Verzweigung, Wiederholung) und Operatoren, die beispielsweise auch Bitoperationen erlauben. Im weiteren Verlauf der Vorlesung werden insbesondere die in C wichtigen maschinennahen Konzepte vorgestellt, wie z.B. Eigenschaften von Datentypen, Arbeiten mit Zeigern, Zeigerarithmetik und Arrays und die verschiedenen Speichersegmente für Code, globale und lokale Daten. Weiterhin wird ein Überblick über die in der C-Standardbibliothek vorhandenen Funktionen und Methoden inklusive der verschiedenen Ansätze für Speicherverwaltung gegeben und ein Einblick in die für die Entwicklung von C-Programmen verwendeten Werkzeuge und deren Zusammenspiel (Compiler, Linker, Makefiles, Unix-Shell) gegeben. Als Abschluß werden typische Sicherheitsprobleme bei der Entwicklung von C-Programmen und deren Konsequenzen vorgestellt sowie Wege aufgezeigt, diese Probleme zu vermeiden.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden sollen dazu in der Lage sein, vorhandenen C-Code zu verstehen und eigenständig C-Programme mittlerer Komplexität modifizieren und grundlegend – mit einem Fokus auf die Erstellung sicherer Software – neu entwickeln zu können. Dabei wird ein besonderer Schwerpunkt auf die Hardwarenähe gelegt, so sollen die Studierenden die in Inf-GRABS-B weiter vertieften Konzepte der Darstellung von Ganz- und Gleitkommazahlen und andere Datenformate wie Strings, die Handhabung von binären Daten, und manuelle Speicherverwaltung verstehen und anwenden können und die Sicherheitsprobleme, die durch manuelle Speicherverwaltung und flexible Typwandlung entstehen können, verstehen und vermeiden können.		
<b>Sonstige Informationen:</b> Das Modul ist insbesondere für Studierende empfohlen, die ihr Studium im Sommersemester beginnen und damit Inf-GRABS-B belegen, ohne vorher Programmierkenntnisse auf Inf-Einf-B erworben zu haben.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> - keine -		
<b>Empfohlene Vorkenntnisse:</b> Parallel zu Inf-GRABS-B		<b>Besondere Bestehensvoraussetzungen:</b> ECTS-Bedingungen de
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Vorlesung Einführung in die C-Programmierung</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Michael Engel <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>1,00 SWS</b> <b>1.5 ECTS</b>
<b>Lernziele:</b> Siehe Modulbeschreibung	

<p><b>Inhalte:</b> Inhalte vgl. Modulbeschreibung</p> <p>Die Vorlesung zu Einführung in die C-Programmierung besteht aus einer Kombination von Präsenzvortrag und Lehrmaterial im flipped classroom-Modus, das die Studierenden sich selbständig aneignen.</p>	
<p><b>Literatur:</b> Jens Gustedt, Modern C, Manning, November 2019, ISBN 9781617295812</p> <p>Brian W. Kernighan, Dennis M. Ritchie: Programmieren in C. 2. Auflage. Hanser, München 1990, ISBN 3-446-15497-3.</p>	
<p><b>Prüfung</b> schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> Im Rahmen der schriftlichen Prüfung sollen die Studierenden nachweisen, dass sie existierenden C-Code lesen und verstehen, Fehler in Code korrigieren und C-Konstrukte selbst erstellen können.</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>Praktische Übungen zu Einführung in die C-Programmierung</b></p> <p><b>Lehrformen:</b> Praktikum, Übung</p> <p><b>Dozenten:</b> Prof. Dr. Michael Engel</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b> Siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b> Im Rahmen der Übungen sollen Studierende anhand gegebener regelmäßiger Aufgaben eigene C-Programme in einer Unix/Linux-Umgebung erstellen, analysieren, und debuggen.</p> <hr/> <p><b>Literatur:</b> Siehe Vorlesung</p>	<p><b>1,00 SWS</b> <b>1.5 ECTS</b></p>

<b>Modul Inf-Projekt1-B Bachelorprojekt 1 der Fächergruppe Informatik</b> <i>Bachelor Project 1 in Computer Science</i>		6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Informatik)		
<b>Inhalte:</b> Das Modul behandelt die praktische Anwendung grundlegender Methoden aus dem Bereich der Informatik im Rahmen eines Projektes. Die behandelten Problemstellungen stammen aus den typischen Fachgebieten der Informatik.		
<b>Lernziele/Kompetenzen:</b> I.d.R aufbauend auf den in den Vorlesungen und Übungen des jeweiligen Faches erworbenen Kenntnissen und Fertigkeiten wird im Projekt eine Forschungs- und/oder Entwicklungsaufgabe mit wissenschaftlichem Bezug in einer Gruppe bearbeitet. Dabei werden die Fähigkeiten im Bereich der Systemanalyse und -entwicklung ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und in der Gruppenarbeit.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorprojekt aus einem der Fachgebiete der Informatik zu wählen. Die wählbaren Projekte sind in UnivIS über das Schlagwort "Inf-Projekt" auffindbar und als Projekte für Bachelorstudierende ausgewiesen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorprojekt 1 der Fächergruppe Informatik</b> <b>Lehrformen:</b> Projektseminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS	<b>4,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben	
<b>Inhalte:</b> Die Inhalte der Bachelorprojekte werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
<b>Literatur:</b>	

---

Die Literatur wird zu Beginn eines Projekts von jedem anbietenden Fachgebiet bekannt gegeben.	
---	--

**Prüfung**

Hausarbeit mit Kolloquium

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen.

Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden von der Betreuerin bzw. dem Betreuer des Projekts zu Beginn der Lehrveranstaltung bekannt gegeben.

<b>Modul Inf-Seminar1-B Bachelorseminar 1 der Fächergruppe Informatik</b> <i>Bachelor Seminar 1 in Computer Science</i>		3 ECTS / 90 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas Weitere Verantwortliche: (qua Amt der bzw. die Studiengangsbeauftragte für den Bachelorstudiengang Informatik)		
<b>Inhalte:</b> Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet der Informatik mit wissenschaftlichen Methoden.		
<b>Lernziele/Kompetenzen:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben. In der Regel sollten zuvor bereits andere Module aus dem Fachgebiet belegt worden sein		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 4.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorseminar 1 der Fächergruppe Informatik</b> <b>Lehrformen:</b> Seminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS	<b>2,00 SWS</b>
<b>Lernziele:</b> wie beim Modul beschrieben	
<b>Inhalte:</b> Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.	
<b>Literatur:</b> Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben	

<b>Prüfung</b> Hausarbeit mit Referat <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen.	
--	--

---

Alternativ kann die Prüfungsleistung auf Hausarbeit mit Kolloquium festgelegt werden. 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.	
---	--

<b>Modul KogSys-GAI-B Genderaspekte in der Informatik</b> <i>Gender Aspects in Computer Science</i>		3 ECTS / 90 h 15 h Präsenzzeit 75 h Selbststudium
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Ute Schmid		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> Das Seminar findet teilweise gemeinsam mit dem Seminar Genderaspekte in der Wirtschaftsinformatik statt, das im Bachelor Wirtschaftsinformatik im Fachgebiet SNA angeboten wird.  Der Arbeitsaufwand von 90 Stunden gliedert sich in etwa wie folgt:  21 Std. Teilnahme an den Präsenzveranstaltungen 24 Std. Literaturarbeit, inklusive Vorbereitung von Kurzpräsentationen 30 Std. Konzeption und Umsetzung des Praxisteils 15 Std. Erstellung der schriftlichen Ausarbeitung		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine Einschränkung		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

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

---

wird in der Veranstaltung bekannt gegeben	
---	--

**Prüfung**

Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Monate

**Beschreibung:**

Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.

<p><b>Modul KogSys-KI-B Einführung in die Künstliche Intelligenz</b>  <i>Introduction to Artificial Intelligence</i></p>	<p>6 ECTS / 180 h</p>
<p>(seit WS24/25)                  Modulverantwortliche/r: Prof. Dr. Ute Schmid</p>	
<p><b>Inhalte:</b>                  Das Modul bietet einen Einstieg in die grundlegende 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 Unsicheres Wissen, Maschinelles Lernen, Sprache und Kommunikation, Bildanalyse, agentenbasierte Ansätze und Robotik werden behandelt. Neben der Vermittlung von theoretischen Grundlagen wird die Umsetzung von KI-Algorithmen in Prolog und Python 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:</p> <ul style="list-style-type: none"> <li>• Problemlösen und Suche</li> <li>• Suchalgorithmen für Spiele</li> <li>• Ansätze der Wissensrepräsentation</li> <li>• Aussagen- und Prädikatenlogik</li> <li>• Inferenz in Logik erster Stufe</li> <li>• Nicht-klassische Logiken</li> <li>• Planung</li> <li>• Maschinelles Lernen</li> <li>• Sprachverarbeitung</li> <li>• Objekt- und Szenenerkennung</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <ul style="list-style-type: none"> <li>• Grundlegende Konzepte und Problemstellungen der KI definieren und erklären können</li> <li>• Einfache KI-Algorithmen auf konkrete – auch neue – Problemstellungen anwenden können</li> <li>• Problemstellungen formal, insbesondere mit Mitteln der Logik modellieren können</li> <li>• Grundzüge von KI-Programmiertechniken (insbesondere funktionale und logische Programmierung) beherrschen</li> </ul>	
<p><b>Sonstige Informationen:</b>                  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:</p> <ul style="list-style-type: none"> <li>• 22,5h Vorlesung + 30h Nachbereitung</li> <li>• 22,5h Übung + 75h Bearbeitung von Übungsaufgaben</li> <li>• 30h Klausurvorbereitung</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>                  (außer für FÜM Kognitive Künstliche Intelligenz)</p>	

<ul style="list-style-type: none"> <li>• Gdl-Mfi-1 (Mathematik für Informatik 1)</li> <li>• DSG-EiAPS-B (Einführung in Algorithmen, Programmierung und Softwaretechnik)</li> </ul>		
<b>Empfohlene Vorkenntnisse:</b> Kenntnisse in den folgenden Bereichen, zugehörige Module in Klammern: <ul style="list-style-type: none"> <li>• Algorithmen und Datenstrukturen (AI-AuD-B)</li> <li>• Introduction to Functional Programming (Gdl-IFP-B oder Gdl-IFP-M)</li> <li>• Grundlagen der Theoretischen Informatik (Gdl-GTI-B)</li> <li>• Lineare Algebra (xAI-MML-M, KTR-Mfi-2-B)</li> </ul>	<b>Besondere Bestehensvoraussetzungen:</b> Keine	
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Einführung in Künstliche Intelligenz</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Ute Schmid <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> Siehe Modulbeschreibung	
<b>Inhalte:</b> Präsentation und Diskussion der Inhalte (siehe Modulbeschreibung), insbesondere theoretische und konzeptionelle Aspekte.	
<b>Literatur:</b> Stuart Russel und Peter Norvig (2021, 4. Auflage). Artificial Intelligence, A Modern Approach (AIMA). Prentice Hall.	
<b>Prüfung</b> schriftliche Modulprüfung (Klausur) / Prüfungsdauer: 105 Minuten <b>Beschreibung:</b> 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. 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: <ul style="list-style-type: none"> <li>• Art und Anzahl der Studienleistungen</li> <li>• Umfang (Anzahl an erreichbaren Punkten) der Studienleistungen</li> <li>• Bearbeitungsdauer der Studienleistungen</li> </ul> Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden. Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialien, Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay.	

Die Aufgabenstellungen in der Klausur sind auf Deutsch und Englisch verfasst.	
<b>Lehrveranstaltungen</b>	
<p><b>Einführung in Künstliche Intelligenz</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Bettina Finzel</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>Siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b></p> <p>Praktische Vertiefungen zu den Inhalten der Vorlesung:</p> <ul style="list-style-type: none"> <li>• Wiederholen und Vertiefen von theoretischen Konzepten, die in der Vorlesung vorgestellt wurden</li> <li>• Simulation von Algorithmen der Suche, der logischen Inferenz, der Planung und des maschinellen Lernens (händisch und programmatisch)</li> <li>• Aufgaben zur Wissensmodellierung und zur Modellierung logischer Welten</li> <li>• Berechnen von Heuristiken</li> <li>• Wahrscheinlichkeitsrechnung</li> <li>• Erarbeiten von Beispielanwendungen in denen Künstliche Intelligenz zum Einsatz kommen kann</li> <li>• Präsentation und Diskussion von Aufgabenlösungen</li> </ul>	<b>2,00 SWS</b>

<b>Modul KogSys-ML-B Einführung in Maschinelles Lernen</b> <i>Introduction to Machine Learning</i>	6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Ute Schmid	
<p><b>Inhalte:</b></p> <p>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.</p> <p>Liste der Themen:</p> <ul style="list-style-type: none"> <li>• Grundkonzepte: Inductive Biases, Hypothesenraum</li> <li>• Evaluation von gelernten Modellen</li> <li>• Entscheidungsbäume und Random Forests</li> <li>• Induktive Logische Programmierung</li> <li>• Künstliche Neuronale Netze</li> <li>• Support Vector Machines, Kernels, Margins</li> <li>• Ausgewählte Ansätze des Deep Learning: Embeddings, Convolutional Neural Networks, Transformer, VAEs</li> <li>• Instanzbasierte Methoden</li> <li>• Bayes'sches Lernen</li> <li>• Lernen von Sequenzen</li> <li>• Reinforcement Learning</li> <li>• Weitere Aspekte: Erklärbarkeit, Neuro-symbolische Ansätze, Knowledge-informed Machine Learning, Human-in-the-loop Learning.</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <ul style="list-style-type: none"> <li>• Grundlegende Konzepte und zentrale Ansätze des maschinellen Lernens erläutern und anwenden können</li> <li>• Zentrale symbolische, neuronale und statistische Algorithmen des Klassifikationslernens auf gegebene Daten anwenden können</li> <li>• Die Eignung gegebener Daten für Algorithmen des Klassifikationslernen beurteilen können</li> <li>• Die Güte gelernter Modelle beurteilen können</li> <li>• Gemeinsamkeiten und Unterschiede zwischen menschlichem und maschinellem Lernen erörtern können</li> <li>• Moderne Bibliotheken für maschinelles Lernen in relevanten Programmiersprachen, insbesondere Python, verwenden können</li> </ul>	
<p><b>Sonstige Informationen:</b></p> <p>Die Folien (Vorlesung und Übung) sowie die Prüfungsunterlagen sind in englischer Sprache verfügbar. Weitere Materialien sind überwiegend in englischer Sprache.</p>	

<p>Der Zeitaufwand gliedert sich in 45h Präsenzzeit und 135h Selbststudium.</p> <p>Zeitaufwand aufgeschlüsselt:</p> <ul style="list-style-type: none"> <li>• 22,5h Vorlesung + 30h Nachbereitung</li> <li>• 22,5h Übung + 75h Bearbeitung von Übungsaufgaben</li> <li>• 30h Klausurvorbereitung</li> </ul>		
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <ul style="list-style-type: none"> <li>• DSG-EiAPS-B (Einführung in Algorithmen, Programmierung und Softwaretechnik)</li> <li>• Gdl-Mfi-1 (Mathematik für Informatik 1)</li> </ul>		
<p><b>Empfohlene Vorkenntnisse:</b></p> <p>Kenntnisse in den folgenden Bereichen, zugehörige Module in Klammern:</p> <ul style="list-style-type: none"> <li>• Objektorientierte Programmierung (DSG-JaP-B, KInf-IPKult-E)</li> <li>• Lineare Algebra (xAI-MML-M, xAI-MML-B, KTR-Mfi-2-B)</li> </ul>		<p><b>Besondere Bestehensvoraussetzungen:</b></p> <p>keine</p>
<p><b>Angebotshäufigkeit:</b> WS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b></p> <p>1 Semester</p>

<p><b>Lehrveranstaltungen</b></p>	
<p><b>1. Lernende Systeme (Machine Learning)</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Dr. Ute Schmid</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>Siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b></p> <p>Präsentation und Diskussion der Inhalte (siehe Modulbeschreibung), insbesondere theoretische und konzeptionelle Aspekte.</p> <hr/> <p><b>Literatur:</b></p> <p>Mitchell, Machine Learning, McGraw-Hill, 1997.</p> <p>Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, 2012.</p> <p>Goodfellow et al., Deep Learning, MIT Press, 2016.</p> <p>Bishop, Pattern Recognition and Machine Learning, Springer, 2006.</p>	<p><b>2,00 SWS</b></p>
<p><b>2. Lernende Systeme (Machine Learning)</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Johannes Langer</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>Siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b></p> <p>Praktische Vertiefungen zu den Inhalten der Vorlesung:</p>	<p><b>2,00 SWS</b></p>

- Wiederholen und Vertiefen von theoretischen Konzepten, die in der Vorlesung vorgestellt wurden
- Implementation von in der Vorlesung vorgestellten Algorithmen
- Handsimulation von in der Vorlesung vorgestellten Algorithmen
- Berechnung von für maschinelles Lernen relevanten Metriken, zur Evaluation oder als Teile von Algorithmen
- Erarbeiten von Beispielanwendungen in denen maschinelles Lernen zum Einsatz kommen kann
- Präsentation und Diskussion von Aufgabenlösungen

**Literatur:**

siehe Vorlesung

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 105 Minuten

**Beschreibung:**

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.

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:

- Art und Anzahl der Studienleistungen
- Umfang (Anzahl an erreichbaren Punkte) der Studienleistungen
- Bearbeitungsdauer der Studienleistungen

Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.

Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialien, Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay.

Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.

<b>Modul MI-EMI-B Einführung in die Medieninformatik</b> <i>Introduction to Media Informatics</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit SS21) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Neben Grundkonzepten der Digitalisierung werden die Medientypen Bild, Audio, Text, Video, 2D-Vektorgrafik sowie 3D-Grafik behandelt. Dabei wird jeweils auf die Erstellung und Bearbeitung entsprechender Medienobjekte sowie deren Kodierung eingegangen.		
<b>Lernziele/Kompetenzen:</b> Studierende sollen zu den verschiedenen Medientypen Beispielformate kennenlernen. Sie sollen die eingesetzten Kompressionsverfahren sowie die dahinter stehenden Philosophien verstehen und die praktischen Einsatzmöglichkeiten einschätzen können. Ferner sollen sie konzeptuelle Kenntnisse und praktische Erfahrungen im Umgang mit Medienobjekten sammeln und z. B. die Erstellung und Bearbeitung von Medientypen wie Text, Bild, Audio und Video selbständig durchführen können.		
<b>Sonstige Informationen:</b> Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb)</li> <li>• Bearbeiten der 3 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb)</li> <li>• Prüfungsvorbereitung und Prüfung: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik (können auch durch den parallelen Besuch eines einführenden Moduls zur Informatik erworben werden)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Einführung in die Medieninformatik</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Andreas Henrich <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen dieser Vorlesung werden nach einer Einführung in das Thema grundlegende Medien und Medienformate betrachtet. Hierzu zählen Bilder, Audio, Texte und Typografie, Video, 2D- und 3D-Grafik.	

Neben den Formaten werden die entsprechenden Grundlagen wie Farbmodelle und Wahrnehmungsmodelle betrachtet. Ziel ist dabei, praktische Fähigkeiten im Umgang mit den genannten Formaten zu vermitteln und die Konzepte von Kodierungs- und Kompressionsverfahren zu erarbeiten. Hierzu geht die Veranstaltung, die einen breiten Überblick über das Gebiet geben soll, an einzelnen ausgewählten Stellen stärker in die Tiefe. Zu nennen sind dabei insbesondere die Medientypen Text, Bild, Audio, Video und 2D-Vektorgrafik.

**Literatur:**

- Malaka, Rainer; Butz, Andreas; Hussmann, Heinrich: Medieninformatik: Eine Einführung. Pearson Studium; 1. Auflage, 2009
- Chapman, Nigel; Chapman Jenny: Digital Multimedia (2nd Edition), John Wiley & Sons, Ltd, 2004
- Henning, Peter A.: Taschenbuch Multimedia , 3. Auflage, Fachbuchverlag Leipzig im Carl Hanser Verlag, 2003
- weitere Literatur wird in der Veranstaltung bekannt gegeben

**2. Einführung in die Medieninformatik**

**Lehrformen:** Übung

**Dozenten:** Mitarbeiter Medieninformatik

**Sprache:** Deutsch

**Angebotshäufigkeit:** WS, jährlich

**2,00 SWS**

**Inhalte:**

Die Inhalte der Vorlesung Einführung in die Medieninformatik werden in den Übungen vertieft und praktisch umgesetzt. Insbesondere werden Kodierungs- und Kompressionsverfahren nachvollzogen, Medienobjekte erstellt und bearbeitet und der Umgang mit einfachen Werkzeugen (z. B. zur Bildbearbeitung) eingeübt.

**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.	
---	--

<b>Modul MI-WAIAI-B Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik</b> <i>Introduction to Academic Research &amp; Writing for Computer Science and Applied Computer Science</i>		3 ECTS / 90 h 22 h Präsenzzeit 68 h Selbststudium
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Im Modul werden wesentliche Methoden und Grundsätze des wissenschaftlichen Arbeitens in Informatik und Angewandter Informatik behandelt: <ul style="list-style-type: none"> <li>• Wissenschaft, Ethik, Forschung</li> <li>• Wissenschaftliche Arbeiten</li> <li>• Wissenschaftliches Arbeiten &amp; Schreiben</li> <li>• Projektmanagement</li> <li>• Vortragen &amp; Präsentieren</li> <li>• Evaluation &amp; empirische Methoden</li> </ul>		
<b>Lernziele/Kompetenzen:</b> 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 wissenschaftlicher Arbeiten unterstützen.  Allgemein werden die Grundlagen zum wissenschaftlichen Arbeiten, zum Verfassen wissenschaftlicher Arbeiten und zum Präsentieren vermittelt und eingeübt.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Kenntnisse der Informatik und Angewandten Informatik.  Das Modul sollte ab dem 3. Semester besucht werden und vor der ersten Belegung eines Seminars.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Andreas Henrich, Mitarbeiter Medieninformatik <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>2,00 SWS</b>
<b>Lernziele:</b> siehe Modulbeschreibung		
<b>Inhalte:</b>		

siehe Modulbeschreibung

**Literatur:**

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

**Prüfung**

Portfolio / Bearbeitungsfrist: 4 Monate

**Beschreibung:**

**Gegenstand** des Portfolios sind alle Inhalte von Vorlesung und Übung.

Im Semester werden studienbegleitend (in der Regel zweiwöchentlich)

Aufgabenstellungen ausgegeben und besprochen, deren individuelle Bearbeitung im Portfolio zu dokumentieren ist.

<b>Modul MII-ROB-B Einführung in die Robotik</b> <i>Introduction to Robotics</i>		6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Markus Rickert		
<b>Inhalte:</b> Das Modul vermittelt einen allgemeinen Überblick in das Gebiet der Robotik. Nach einer Einführung in grundlegende mathematische Konzepte wird auf die Modellierung von Robotersystemen sowie auf die Berechnung von Kinematik und Dynamik eingegangen. Weitere Aspekte beinhalten die Regelung von Robotersystemen, die Berechnung von Trajektorien und Verfahren zur Bahnplanung und Kollisionsvermeidung. Darüber hinaus werden Grundlagen der Roboterprogrammierung und aktuelle Softwarearchitekturen für Robotersysteme behandelt.		
<b>Lernziele/Kompetenzen:</b> Studierende lernen grundlegende Konzepte der Robotik und deren Anwendung.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Kenntnisse in Informatik, Kenntnisse in Mathematik und linearer Algebra, sowie Programmierkenntnisse (C++).		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> Semester
<b>Lehrveranstaltungen</b>		
<b>1. Einführung in die Robotik</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Markus Rickert <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich		<b>2,00 SWS</b>
<b>Literatur:</b> John J. Craig, Introduction to Robotics: Mechanics and Control, Pearson. Steven M. LaValle, Planning Algorithms, Cambridge University Press.		
<b>2. Einführung in die Robotik</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich		<b>2,00 SWS</b>
<b>Lernziele:</b> In der Übung werden die Inhalte der Vorlesung vertieft und deren praktische Anwendung geübt.		
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b>		

---

Die Prüfungsform (schriftlich oder mündlich) wird zu Beginn der Lehrveranstaltung bekannt gegeben.	
--	--

<b>Modul MOBI-DBS-B Datenbanksysteme</b> <i>Database Systems</i>		6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> Das Modul vermittelt eine systematische Einführung in das Gebiet der Datenbanksysteme.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Datenbanksysteme</b> <b>Lehrformen:</b> Vorlesung, Übung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS		4,00 SWS
<b>Lernziele:</b> 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.		
<b>Inhalte:</b> <ul style="list-style-type: none"> <li>• Datenbank-Konzepte und -Architektur</li> <li>• Modellierung von Datenbanken: Das ER- und EER-Modell</li> <li>• Das relationale Modell</li> <li>• Relationale Algebra</li> <li>• SQL (DDL und DML)</li> </ul>		

<ul style="list-style-type: none"> <li>• Normalisierung und Normalformen</li> <li>• Datenbanken im Mehrbenutzerbetrieb: Transaktionssysteme und Recovery</li> <li>• Alternative Entwicklungen im Bereich Datenbanken</li> </ul>	
<p><b>Literatur:</b>  Date C.J.: An Introduction to database systems. 8th Edition, Addison-Wesley, Reading, Massachusetts 2003  Elmasri &amp; Navathe: Grundlagen von Datenbanksystemen, Pearson, 2002</p>	
<p><b>Prüfung</b>  schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 105 Minuten</p> <p><b>Beschreibung:</b>  Dezentrale Prüfung. Gegenstand der Prüfung sind alle Inhalte von Vorlesung und Übung (einschließlich Teilleistungen; siehe unten).</p> <p>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.</p> <p>Im Semester werden studienbegleitend Teilleistungen ausgegeben und besprochen, deren Abgabe freiwillig ist. Für die Bearbeitung dieser Teilleistungen können Bonuspunkte vergeben werden. Die Anzahl und Bedingungen der zu erreichenden Bonuspunkte sowie deren Umrechnungsfaktor in mögliche Klausurpunkte werden in der ersten Übungsstunde bekannt gegeben.</p> <p>Ist die Prüfung 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.</p>	

<b>Modul MOBI-DE-B Data Engineering</b> <i>Data Engineering</i>		6 ECTS / 180 h 30 h Präsenzzeit 150 h Selbststudium
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> <ul style="list-style-type: none"> <li>• Grundlagen und Konzepte des Data Engineering</li> <li>• Dateninfrastrukturen und -verwaltung</li> <li>• Umgang mit Datenqualität und -konsistenz</li> <li>• Umgang mit Heterogenität unterschiedlicher Datenquellen</li> <li>• Data Lakes und ihre Anwendungen</li> <li>• Praktische Übungen</li> </ul>		
<b>Lernziele/Kompetenzen:</b> Die Studierenden kennen die zentralen Herausforderungen der Datenbereitstellung für statistische Analysen und das maschinelle Lernen sowie praxisrelevante Lösungsansätze und sind in der Lage, diese anzuwenden.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Programmierkenntnisse in Python, Grundlegende Kenntnisse in SQL  Vorleistung: Vormodul (I Inf-Einf-B) - empfohlen  Modul Datenbanksysteme (MOBI-DBS-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> ECTS-Bedingungen de
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Data Engineering</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>4,00 SWS</b>
<b>Prüfung</b> Hausarbeit mit Kolloquium, MOBI-DE-B / Prüfungsdauer: 2 Monate		

<b>Modul NLProc-ALV-B Algorithmisches Sprachverstehen</b> <i>Natural Language Understanding</i>		6 ECTS / 180 h
(seit SS24) Modulverantwortliche/r: Prof. Dr. Roman Klinger		
<b>Inhalte:</b> The lecture on natural language processing covers a set of topics: <ul style="list-style-type: none"> <li>• Overview Natural Language Understanding</li> <li>• Lexical Semantics</li> <li>• Distributional Semantics</li> <li>• Word Embeddings</li> <li>• Contextualized Representations</li> <li>• Information Extraction</li> <li>• Semantic Role Labeling</li> <li>• Argument Mining</li> <li>• Sentiment Analysis</li> <li>• Natural Language Inference and Language Models for Instruction Answering</li> </ul>		
<b>Lernziele/Kompetenzen:</b> The student understands challenges and tasks in natural language understanding and acquired a basic knowledge of methods to approach and solve them.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> The lecture on natural language understanding loosely builds on top of the Information Retrieval and Text Mining (IRTM) lecture. In the IRTM lecture, we learned how to handle text documents, search in them, classify them, and group them. We also looked into machine learning methods that help us organizing them. In natural language understanding, we look deeper into text and study the content on the word, phrase, sentence or paragraph level. However, it will also be possible to follow this lecture without having attended IRTM. Prior knowledge in programming is, however, required.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 4.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Einführung in das Algorithmische Sprachverstehen</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Roman Klinger <b>Sprache:</b> Deutsch / English on demand/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>4,00 SWS</b>
<b>Inhalte:</b> The lecture on natural language processing covers a set of topics: <ul style="list-style-type: none"> <li>• Overview Natural Language Understanding</li> </ul>	

<ul style="list-style-type: none"><li>• Lexical Semantics</li><li>• Distributional Semantics</li><li>• Word Embeddings</li><li>• Contextualized Representations</li><li>• Information Extraction</li><li>• Semantic Role Labeling</li><li>• Argument Mining</li><li>• Sentiment Analysis</li><li>• Natural Language Inference and Language Models for Instruction Answering</li></ul>	
<b>Literatur:</b> Daniel Jurafsky and James Martin, Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition	
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten	

<b>Modul NLProc-IRTM-B Information Retrieval and Text Mining</b>		6 ECTS / 180 h
<i>Information Retrieval and Text Mining</i>		
(seit SS24)		
Modulverantwortliche/r: Prof. Dr. Roman Klinger		
<b>Inhalte:</b>		
<ul style="list-style-type: none"> <li>• Boolean retrieval, inverted index</li> <li>• Wild card queries, tolerant retrieval, spelling correction, query expansion</li> <li>• Tokenization, Term normalization, Term statistics</li> <li>• Efficient storage, indexing, compression, and memory consumption estimates</li> <li>• Evaluation</li> <li>• Ranking, Cosine similarity, TFIDF, Language models, Probabilistic retrieval</li> <li>• Text classification, naive Bayes, SVM, MaxEnt Classifier, Neural Networks</li> <li>• Flat and hierarchical clustering</li> <li>• Web analysis, Page Rank</li> </ul>		
<b>Lernziele/Kompetenzen:</b>		
Students learn how to build a search engine for text and evaluate it with various features. They learn to classify documents and group them according to their content. Students understand both the theoretical background of the methods and models and learn how to apply them.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
The course develops a fundamental understanding of handling textual documents and large document collections. Knowledge of one higher programming language is strongly recommended, but not essential.		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<b>Information Retrieval and Text Mining</b>		<b>4,00 SWS</b>
<b>Lehrformen:</b> Vorlesung und Übung		
<b>Dozenten:</b> Prof. Dr. Roman Klinger		
<b>Sprache:</b> Englisch/Deutsch / English on demand		
<b>Angebotshäufigkeit:</b> SS, jährlich		
<b>Lernziele:</b>		
Students learn how to build a search engine for text and evaluate it with various features. They learn to classify documents and group them according to their content. Students understand both the theoretical background of the methods and models and learn how to apply them.		
<b>Inhalte:</b>		
<ul style="list-style-type: none"> <li>• Boolean retrieval, inverted index</li> <li>• Wild card queries, tolerant retrieval, spelling correction, query expansion</li> <li>• Tokenization, Term normalization, Term statistics</li> </ul>		

<ul style="list-style-type: none"><li>• Efficient storage, indexing, compression, and memory consumption estimates</li><li>• Evaluation</li><li>• Ranking, Cosine similarity, TFIDF, Language models, Probabilistic retrieval</li><li>• Text classification, naive Bayes, SVM, MaxEnt Classifier, Neural Networks</li><li>• Flat and hierarchical clustering</li><li>• Web analysis, Page Rank</li></ul>	
<b>Literatur:</b> Introduction to Information Retrieval, Manning, Raghavan, Schütze.	
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten	

<b>Modul PSI-DatSchu-B Datenschutz</b>		3 ECTS / 90 h
<i>Data Protection</i>		
(seit SS23) Modulverantwortliche/r: N.N. Weitere Verantwortliche: Prof. Dr. Dominik Herrmann		
<b>Inhalte:</b> 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		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine; alle notwendigen Inhalte werden in der Veranstaltung erarbeitet.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester Semester
<b>Lehrveranstaltungen</b>		
<b>Datenschutz</b> <b>Lehrformen:</b> Vorlesung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>2,00 SWS</b>
<b>Lernziele:</b> siehe Modulbeschreibung		
<b>Literatur:</b> 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		
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten		

<b>Modul PSI-EDS-B Ethics for the Digital Society</b> <i>Ethics for the Digital Society</i>		3 ECTS / 90 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Dominik Herrmann		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> 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.  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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Ethics for the Digital Society</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Dominik Herrmann <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> cf. module description	
<b>Inhalte:</b> cf. module description	
<b>Literatur:</b> <ul style="list-style-type: none"> <li>Ibo van de Poel and Lamber Royackers: Ethics, Technology, and Engineering – an Introduction</li> </ul>	

- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>• Jay Quinn: Ethics for the Information Age</li><li>• Herman T. Tavani: Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing</li></ul> |  |
|---|--|

**Prüfung**

schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 80 Minuten

**Beschreibung:**

The exam time includes a reading time of 20 minutes.

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.

The maximum number of points that can be achieved in the exam is 100.

Participants that solve assignments can collect bonus points. Details regarding the total number of bonus points, the number of assignments, the number of points per assignment, and the type of assignments will be announced in the first lecture.

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.

<b>Modul PSI-IntroSP-B Introduction to Security and Privacy</b> <i>Introduction to Security and Privacy</i>	6 ECTS / 180 h
(seit WS24/25) Modulverantwortliche/r: Prof. Dr. Dominik Herrmann	
<b>Inhalte:</b> 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.	
<b>Lernziele/Kompetenzen:</b> 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.	
<b>Sonstige Informationen:</b> 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.  Workload breakdown: <ul style="list-style-type: none"> <li>• Lecture: 22.5 hours (2 hours per week)</li> <li>• Tutorials: 22.5 hours (2 hours per week)</li> <li>• Preparation and studying during the semester: 30 hours</li> <li>• Assignments: 67.5 hours</li> <li>• Preparation for the exam (including the exam itself): 37.5 hours</li> </ul>	
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine	
<b>Empfohlene Vorkenntnisse:</b> 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.  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.  Moreover, basic familiarity with common web technologies (HTTP, HTML, JavaScript) as well as relational database systems and SQL is a recommended prerequisite.	<b>Besondere Bestehensvoraussetzungen:</b> keine

<p>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.</p> <p>Modul Einführung in die Informatik (Inf-Einf-B) - empfohlen                  Modul Grundlagen der Rechnerarchitektur und Betriebssysteme (Inf-GRABS-B) - empfohlen</p>		
<p><b>Angebotshäufigkeit:</b> WS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b> 1 Semester</p>

<b>Lehrveranstaltungen</b>	
<p><b>1. Introduction to Security and Privacy</b>  <b>Lehrformen:</b> Vorlesung  <b>Sprache:</b> Englisch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b>                      cf. module description</p> <hr/> <p><b>Inhalte:</b>                      Selected topics</p> <ul style="list-style-type: none"> <li>• Security Terminology (protection goals, attacker and attack types)</li> <li>• Authentication and Authorization Fundamentals</li> <li>• Software Security in C and Assembly (e.g., buffer overflows, selected defenses)</li> <li>• Cryptography (e.g., historic ciphers, symmetric and asymmetric cryptosystems, Diffie-Hellman key exchange, TLS protocol)</li> <li>• Network Security (spoofing, denial of service, authentication protocols, intrusion detection systems)</li> <li>• Web Security (attacks and defenses related to the OWASP Top 10 including SQL injections and Cross Site Scripting)</li> <li>• Privacy and Techniques for Data Protection (re-identification risks, anonymization networks, k-anonymity, the idea of differential privacy)</li> </ul> <hr/> <p><b>Literatur:</b>                      Selected books:</p> <ul style="list-style-type: none"> <li>• A. Shostack: Threat Modelling</li> <li>• W. Stallings: Computer Security: Principles and Practice</li> <li>• J. Erickson: Hacking: The Art of Exploitation</li> </ul>	<p><b>2,00 SWS</b></p>
<p><b>2. Introduction to Security and Privacy</b>  <b>Lehrformen:</b> Übung  <b>Sprache:</b> Englisch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>                      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.</p>	<p><b>2,00 SWS</b></p>

**Prüfung**

Testat / Prüfungsdauer: 90 Minuten

**Beschreibung:**

In the intermediate examination (e-exam), participants demonstrate that they master the practical skills acquired by completing the assignments.

**Prüfung**

schriftliche Prüfung (E-Prüfung) / Prüfungsdauer: 120 Minuten

**Zulassungsvoraussetzung zur Modulprüfung:**

To be admitted to the final examination (e-exam), participants must have passed the intermediate exam (e-exam).

**Beschreibung:**

The exam time includes a reading time of 30 minutes.

Details about the requirements for admission to the written examination will be announced in the first lecture.

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.

<b>Modul SWT-FSE-B Foundations of Software Engineering</b> <i>Foundations of Software Engineering</i>		6 ECTS / 180 h
(seit SS23) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> Students will receive an introduction to the common problems and paradigms in, and foundations of, software development. They will also gather conceptual and practical knowledge in the analysis, design and testing of software, with an emphasis on technical aspects of specifying, designing, implementing, verifying and validating software.		
<b>Sonstige Informationen:</b> The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German.  The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> <li>• 45 hrs. attending lectures (Vorlesungen)</li> <li>• 30 hrs. reviewing the lectures, including researching and studying material from additional sources</li> <li>• 45 hrs. attending practicals (Übungen)</li> <li>• 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources</li> <li>• 30 hrs. preparing for the written exam (Klausur)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in Computer Science, as well as knowledge in programming in Java and in algorithms and data structures.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Foundations of Software Engineering</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>3,00 SWS</b>
<b>Inhalte:</b> 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**

**Lehrformen:** Übung

**Dozenten:** Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen

**Sprache:** Englisch/Deutsch

**Angebotshäufigkeit:** SS, jährlich

**3,00 SWS**

**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.

<b>Modul SYSNAP-SNAP-B Systemnahe Programmierung</b> <i>Systems Programming</i>	6 ECTS / 180 h
(seit WS25/26) Modulverantwortliche/r: Prof. Dr. Michael Engel	
<p><b>Inhalte:</b></p> <p><b><i>Das Modul wird voraussichtlich erst ab dem WS 2026/27 angeboten!</i></b></p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>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.</p> <p>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.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>Bestandene Prüfung in Inf-GRABS-B oder PSI-EiRBS-B</p>	

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

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

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

<b>Modul SuStat-013-M Introduction to Econometrics</b> <i>Introduction to Econometrics</i>		6 ECTS / 180 h
(seit WS23/24) Modulverantwortliche/r: Prof. Dr. Timo Schmid		
<b>Inhalte:</b> Vermittlung grundlegender Kenntnisse in Verfahren der modernen Ökonometrie, insbesondere Analyseverfahren auf Basis der Methode der kleinsten Quadrate für abhängige stetige Variablen.		
<b>Lernziele/Kompetenzen:</b> Im Rahmen dieses Moduls sollen grundlegende Kenntnisse in Verfahren der modernen Ökonometrie sowie die Fähigkeit erworben werden, statistische Methoden richtig anzuwenden, zu bewerten und ihre Ergebnisse zu interpretieren. Im Mittelpunkt stehen dabei Anwendungsmöglichkeiten sowie Grenzen von Verfahren mittels der Kleinst-Quadrate-Methode. Diskutiert werden klassische lineare Regressionsmodelle.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b> Modul Methoden der Statistik I (Stat-B-01) - Modul Methoden der Statistik II (Stat-B-02) -		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Introduction to Econometrics</b> <b>Lehrformen:</b> Vorlesung, Übung <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>4,00 SWS</b>

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls.	
---	--

<b>Modul SuStat-014-M Advanced Econometrics</b> <i>Advanced Econometrics</i>		6 ECTS / 180 h
(seit WS23/24) Modulverantwortliche/r: Prof. Dr. Timo Schmid		
<b>Inhalte:</b> Vermittlung grundlegender Kenntnisse in Verfahren der modernen Ökonometrie, insbesondere Analyse von Querschnittsdaten mittels der Maximum-Likelihood-Methode z.B. für abhängige binäre oder begrenzt stetige Variablen.		
<b>Lernziele/Kompetenzen:</b> Im Rahmen dieses Moduls sollen grundlegende Kenntnisse in Verfahren der modernen Ökonometrie sowie die Fähigkeit erworben werden, statistische Methoden richtig anzuwenden, zu bewerten sowie ihre Ergebnisse zu interpretieren. Im Mittelpunkt stehen dabei Anwendungsmöglichkeiten sowie -grenzen von Verfahren mittels der Maximum-Likelihood-Methode. Diskutiert werden verallgemeinerte Regressionsmodelle für binäre oder geordnete Variablen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b>  Modul Grundlagen der Ökonometrie (SuStat-013-M) - Modul Methoden der Statistik III (SuStat-036-M) -		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Advanced Econometrics</b> <b>Lehrformen:</b> Vorlesung, Übung <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>4,00 SWS</b>
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls.		

<b>Modul SuStat-071-M Advanced Data Analysis With R</b> <i>Advanced Data Analysis With R</i>		6 ECTS / 180 h
(seit WS23/24) Modulverantwortliche/r: Prof. Dr. Timo Schmid		
<b>Inhalte:</b> Anwendung fortgeschrittener Methoden mit der Statistiksoftware R.		
<b>Lernziele/Kompetenzen:</b> In diesem Modul soll fortgeschrittene Datenanalyse mit der Statistiksoftware R vermittelt werden. Neben der Vermittlung grundlegender Techniken wie verallgemeinerten linearen Modellen werden dabei auch Themen wie die Visualisierung von (hochdimensionalen) Daten und Data-Mining-Methoden behandelt.		
<b>Sonstige Informationen:</b> Ab dem WS 2023/24 umfasst dieses Modul 6 ECTS-Punkte.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Kenntnisse der Programmiersprache R. Modul Methoden der Statistik I (Stat-B-01) - Modul Methoden der Statistik II (Stat-B-02) -		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Advanced Data Analysis With R</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>

<b>Prüfung</b> schriftliche Hausarbeit / Bearbeitungsfrist: 8 Wochen <b>Beschreibung:</b> Die jeweils gültige Prüfungsform sowie Prüfungssprache wird zu Beginn der Veranstaltung durch den Veranstalter bekannt gegeben. Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls.	
---	--

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> Die jeweils gültige Prüfungsform sowie Prüfungssprache wird zu Beginn der Veranstaltung durch den Veranstalter bekannt gegeben. Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls.	
--	--



<b>Modul SuStat-075-M Statistische Programmierung mit R</b> <i>Statistical Programming with R</i>		6 ECTS / 180 h
(seit WS23/24) Modulverantwortliche/r: Prof. Dr. Timo Schmid		
<b>Inhalte:</b> Fortgeschrittene Anwendung und Programmierung mit der Statistiksoftware R, u.a. Programmierung von Schleifen und Funktionen.		
<b>Lernziele/Kompetenzen:</b> Im Rahmen dieses Grundlagenmoduls soll die Umsetzung der bereits erworbenen statistischen und ökonometrischen Kenntnisse mit der Statistiksoftware R vertieft und eine Einführung in das Programmieren mit R vermittelt werden.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine.		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Kenntnisse der Programmiersprache R.		<b>Besondere Bestehensvoraussetzungen:</b> keine.
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> Semester

<b>Lehrveranstaltungen</b>	
<b>Statistische Programmierung mit R</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS	<b>2,00 SWS</b>

<b>Prüfung</b> schriftliche Hausarbeit / Bearbeitungsfrist: 8 Wochen	
---	--

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

<b>Modul SuStat-076-M Fortgeschrittene Statistik</b> <i>Advanced Statistics</i>		6 ECTS / 180 h
(seit WS23/24) Modulverantwortliche/r: Prof. Dr. Christian Aßmann		
<b>Inhalte:</b> Vermittlung der Grundlagen der Statistischen Theorie, insbesondere der Wahrscheinlichkeitstheorie, parametrischer Verteilungsfamilien, Asymptotik, sowie Transformations- und Faltungssätze.		
<b>Lernziele/Kompetenzen:</b> Im Rahmen dieses Kurses werden die Studierenden mit der Anwendung grundlegender statistischer Methoden vertraut gemacht. Darüber hinaus werden theoretische Grundlagen der statistischen Theorie vermittelt. Diese theoretischen Grundlagen versetzen die Studierenden in die Lage, sich weitergehende Ergebnisse der Statistischen Theorie selbständig anzueignen. Im Mittelpunkt des Kurses stehen dabei neben den fundamentalen Theoremen der Wahrscheinlichkeitstheorie, Eigenschaften parametrischer Verteilungsfamilien, Grundlagen der asymptotischen Theorie, sowie Faltungs- und Transformationsätze.		
<b>Sonstige Informationen:</b> Nachfolgemodul zu SuStat-036-M "Methoden der Statistik III".		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b>  Modul Methoden der Statistik I (Stat-B-01) - Modul Methoden der Statistik II (Stat-B-02) -		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b> 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Fortgeschrittene Statistik</b> <b>Lehrformen:</b> Vorlesung, Übung <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich		<b>3,00 SWS</b>
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls. Die jeweils gültige Prüfungssprache wird zu Beginn der Veranstaltung durch den Veranstalter bekannt gegeben.		

<b>Modul SuStat-079-M Analyse hochdimensionaler Daten</b> <i>Analysis of High-dimensional Data</i>		6 ECTS / 180 h
(seit WS23/24 bis WS25/26) Modulverantwortliche/r: Prof. Dr. Timo Schmid		
<b>Inhalte:</b> Multivariate Verfahren wie Varianzanalyse, Distanzmaße, Hauptkomponentenanalyse, Faktorenanalyse und Clusteranalyse, Latente Klassenanalyse sowie Klassifikations- und Regressionsbäume.		
<b>Lernziele/Kompetenzen:</b> Die Analyse von komplexen Datensätzen benötigt unterschiedliche multivariate Verfahren, welche in diesem Modul vorgestellt werden sollen. Fragestellungen und Anwendungsfälle, dargestellt an einfachen Beispielen, sollen die Konzepte des (Un-)Supervised Learning Schritt für Schritt verdeutlichen.		
<b>Sonstige Informationen:</b> Nachfolgemodul zu SuStat-033-M "Multivariate Verfahren".		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Keine		
<b>Empfohlene Vorkenntnisse:</b> Modul Methoden der Statistik I (Stat-B-01) - Modul Methoden der Statistik II (Stat-B-02) -		<b>Besondere Bestehensvoraussetzungen:</b> Keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Analyse hochdimensionaler Daten</b> <b>Lehrformen:</b> Vorlesung, Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>3,00 SWS</b>
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Prüfung beinhaltet Fragen und Aufgaben zu den Studieninhalten des Moduls.		

<b>Modul UxD-G-M Grundlagen des Gestaltens</b> <i>Fundamentals of Design</i>		6 ECTS / 180 h
(seit SS25) Modulverantwortliche/r: Prof. Dr. Patrick Tobias Fischer		
<p><b>Inhalte:</b></p> <p>Dieses Modul bereitet Studierende für die Handlungsfähigkeit im Bereich der Forschung durch Design vor. Dabei soll vor allem die tatkräftige Gestaltungsfähigkeit ausgebildet werden, indem eine breite Wissens- und Erfahrungsbasis von gestaltungspraktischen, theoretischen, wissenschaftlichen und technischen Kenntnissen und Fertigkeiten geschaffen wird.</p> <p>Die vermittelten Grundlagen sind unabhängig von konkreten Anwendungserfordernissen und dienen dem Aufbau eines persönlichen gestalterischen Repertoires von Methoden, Phänomenen, Techniken und formalästhetischen Erfahrungen für zukünftige Gestaltungsaufgaben im Forschungsprozess.</p> <p>Folgende Themenfelder werden behandelt: Zeichnen, Grundlagen der Komposition, Wort, visuelle Repräsentation, Licht, Objekt, Prototyping, Raum und Zeit.</p> <p>Durch eine Vielzahl von Kompositionsaufgaben werden die verschiedenen Themenfelder mit analogen und digitalen Werkzeugen (Processing, TouchDesigner, etc.) bearbeitet und exploriert, um somit die eigene Sprachfähigkeit für Oberflächen, Körper und Räume herzustellen sowie Gestaltseinheit und -höhe gezielt komponieren zu können.</p>		
<p><b>Lernziele/Kompetenzen:</b></p> <p>Im Rahmen dieses Moduls sollen sich Studierende eine grundlegende gestalterische Erfahrungsbasis erarbeiten. Die erworbenen Fachkenntnisse umfassen dabei Methodenkenntnisse, Prinzipien, Konzepte und Arbeitsweisen. Diese fachlichen Kompetenzen befähigen Studierende das eigene erworbene gestalterische Repertoire für den wissenschaftlichen Forschungsprozess (Forschung durch Gestaltung) zu verwenden.</p> <p>Darüber hinaus sind die Grundlagen des Gestaltens auf viele weitere Kontexte übertragbar, indem das eigene Abstraktionsvermögen durch praktische Übungen erweitert wird. Weiterhin ermöglicht diese Kompetenz bereits bestehendes Wissen als Grundlage für die Entwicklung eigener Ideen in den anwendungsorientierten Übungen zu verfestigen. Durch die Wissensverbreiterung um den gestalterischen Bereich werden hierbei neue Tätigkeitsfelder erschlossen, da grafische, herstellungstechnische und räumliche Fachbegriffe benannt und erläutert werden können.</p> <p>Absolvent:innen sind in der Lage ihr gestalterisches Repertoires selbständig zu erweitern, indem sie Formen und Herstellungstechniken analysieren und vergleichen; Entscheidungen und Entwürfe beurteilen und begründen; konkrete Designbeispiele und -ergebnisse analysieren und diskutieren.</p>		
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine</p>		
<p><b>Empfohlene Vorkenntnisse:</b> Voraussetzungen für dieses Modul sind ein gewisses Maß an Gestaltungsaffinität sowie Kritik- und Teamfähigkeit. Eine Diskussionsbereitschaft wird erwartet.  Grundlegende Programmierkenntnisse sollten vorhanden sein.</p>		<p><b>Besondere Bestehensvoraussetzungen:</b> keine</p>
<p><b>Angebotshäufigkeit:</b> SS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b> 1 Semester</p>

<b>Lehrveranstaltungen</b>	
<p><b>1. Grundlagen des Gestaltens</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Dr. Patrick Tobias Fischer</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>Bildung eines gestalterischen Repertoires und die Entwicklung gestalterischen Denkens. Die Themen strukturieren sich hierbei nach den für die Interaktionsgestaltung wesentlichen Dimensionen: Wort, visuelle Repräsentation, Objekte und Raum, Zeit und Verhalten. Die Themenfelder werden jeweils durch eigene gestalterische Auseinandersetzung mit analogen und digitalen Medien begleitet.</p> <hr/> <p><b>Inhalte:</b></p> <p>Inhalt:</p> <p>Zeichnen</p> <ul style="list-style-type: none"> <li>- Zeichenwerkzeuge, Malwerkzeuge</li> <li>- Bildgebende Techniken</li> <li>- Handgemachte Drucktechniken</li> <li>- Industrielle Drucktechniken</li> <li>- Zeichenhilfen</li> <li>- Die Relevanz von Skizzen</li> </ul> <p>Grundlagen der Komposition</p> <ul style="list-style-type: none"> <li>- Kompositionslehre</li> <li>- Proportionslehre und -definitionen</li> <li>- Darstellungstechniken</li> <li>- Formwirkungen</li> </ul> <p>Words</p> <ul style="list-style-type: none"> <li>- Schrift</li> <li>- Layout</li> </ul> <p>Visuelle Repräsentation</p> <ul style="list-style-type: none"> <li>- Ornament, Muster</li> <li>- Zeichen, Logo</li> </ul> <p>Licht</p> <ul style="list-style-type: none"> <li>- Farbenwahrnehmung</li> <li>- Farbmischung, Farbsysteme</li> <li>- Farbordnungen, Farbtheorien</li> <li>- Farbkontraste</li> </ul>	<p><b>2,00 SWS</b></p>

- Farbwirkungen

#### Objekt

- Formfindung
- Die dritte Dimension
- Formenvielfalt
- Formenaussage
- Formenbeziehung
- Prinzipien der Formgliederung
- Formunterscheidung
- Materialität

#### Prototyping

- Fidelity
- Designspaces and -process
- Materialien & Fertigungsverfahren: Subtraktive und additive

Fertigungsmethoden, Urformen (Gussverfahren), Umformen (Vacuum Forming, etc.)

#### Raum

- Begriffsklärung
- Place Qualifying Agents
- Space-establishing Elements
- Raum und Form
- Formqualitäten
- Gleichzeitigkeit von Räumen
- Verbindung von Räumen

#### Zeit

- Storyboarding
- Animation
- Arten der zeitlichen Gliederung
- Rhythmus und Momentum
- Transitions

---

#### Literatur:

Waeger, Markus, 2014. Grafik und Gestaltung.

Forssmann, Friedrich und Ralf DE JONG, Detailtypografie. Nachschlagewerk für alle Fragen zu Schrift und Satz.

Henkel, Katharina, 2012. Zwischen Film und Kunst: Storyboards von Hitchcock bis Spielberg. Emden: Kunsthalle Emden.

<p>Garret, Jesse James, 2011. The elements of user experience: user-centered design for the web and beyond. 2. Auflage. Berkeley, Calif.: New Riders.</p> <p>Krohn, Michael und Burg GIEBICHENSTEIN, 2010. Formfächer: Design - Begriffe - Begreifen. Ludwigsburg: av-Ed..</p> <p>Bürdek, Bernhard E., 2015. Design: Geschichte, Theorie und Praxis der Produktgestaltung. 4. Auflage. Basel: Birkhäuser.</p>	
<p><b>2. Grundlagen des Gestaltens</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Patrick Tobias Fischer</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>Aus der praktischen Auseinandersetzung mit den Themenfeldern Wort, visuelle Repräsentation, Objekt, Raum, Zeit und Verhalten entsteht ein Verständnis über allgemeine Phänomene der Gestaltung und bereitet Lernende für einen weit gefassten Diskurs vor, der aus Fächer- und Fachbereichsgrenzen hinaus weist und vor allem darstellende Fähigkeiten mittels verschiedenster Werkzeuge und Techniken einübt.</p> <hr/> <p><b>Inhalte:</b></p> <p>Inhalt:</p> <p>In den thematischen Entwurfsübungen kommen verschiedenste Werkzeuge zur Anwendung, wie z.B.</p> <ul style="list-style-type: none"> <li>• Processing</li> <li>• Unity, Blender</li> <li>• TouchDesigner</li> </ul>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b></p> <p>Portfolio / Prüfungsdauer: 140 Stunden</p>	

<b>Modul VIS-GIV-B Grundlagen der Informationsvisualisierung</b> <i>Foundations of Information Visualization</i>		6 ECTS / 180 h
(seit SS22) Modulverantwortliche/r: Prof. Dr. Fabian Beck		
<b>Inhalte:</b> 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: <ul style="list-style-type: none"> <li>• Numerische Daten (Diagramme für univariate Verteilungen, multivariate Daten und Zeitreihen)</li> <li>• Kategoriale Daten (Mengen- und Ereignisvisualisierungen)</li> <li>• Relationale Daten (Visualisierungen für Graphen und Hierarchien)</li> <li>• Räumlich-zeitliche Daten (Visualisierung von Bewegung sowie räumlich zugeordnete Zeitreihen und Ereignisse)</li> </ul> Unterstützende Werkzeuge und Technologien für die Erstellung solcher Visualisierungen werden ebenfalls vorgestellt und genutzt.		
<b>Lernziele/Kompetenzen:</b> 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.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Präsenzzeit in Vorlesung und Übung: 45h</li> <li>• Vor- und Nachbereitung der Vorlesung: 30h</li> <li>• Bearbeitung von Übungen und Studienleistungen: 75h</li> <li>• Vorbereitung zur Prüfung: 30h</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Programmierkenntnisse; Algorithmen und Datenstrukturen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> ab dem 3.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Grundlagen der Informationsvisualisierung</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Fabian Beck <b>Sprache:</b> Deutsch		<b>2,00 SWS</b>

<p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> Siehe Modulbeschreibung</p> <hr/> <p><b>Literatur:</b> Weiterführende Unterlagen werden in der Veranstaltung bekannt gegeben.</p>	
<p><b>2. Grundlagen der Informationsvisualisierung</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> N.N.</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> In der Übung werden Vorlesungsinhalte vertieft und deren praktische Anwendung geübt.</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> 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.</p>	

<b>Modul WI-Projekt-B Bachelorprojekt aus der Fächergruppe Wirtschaftsinformatik</b> <i>Bachelor Project in Information Systems</i>		6 ECTS / 180 h
(seit WS21/22) Modulverantwortliche/r: Prof. Dr. Sven Overhage		
<b>Inhalte:</b> In einem Projekt werden spezifische Fragestellungen aus Teilgebieten der Wirtschaftsinformatik bearbeitet und diskutiert.		
<b>Lernziele/Kompetenzen:</b> Die Veranstaltung bereiten auch auf das systematische Arbeiten im Team vor und fördern Schlüsselqualifikationen wie die Präsentation von Arbeitsergebnissen und die zielgerichtete Bearbeitung von Projekten.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Bachelorprojekt aus der Fachgruppe Wirtschaftsinformatik</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS	<b>4,00 SWS</b>

<b>Prüfung</b> Hausarbeit mit Kolloquium <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung. <b>Beschreibung:</b> Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen. Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquium werden von der Betreuerin bzw. dem Betreuer der Projektarbeit zu Beginn der Lehrveranstaltung bekannt gegeben.	
--	--

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

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

<b>Prüfung</b> Hausarbeit mit Referat <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Als Prüfungsleistung ist eine Hausarbeit sowie ein Referat zu erbringen. Alternativ kann die Prüfungsleistung auf Hausarbeit mit Kolloquium festgelegt werden. 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.	
--	--

<b>Modul WiMa-B-001 Wirtschaftsmathematik: Lineare Algebra</b>		6 ECTS / 180 h
<i>Mathematics for Economics and Business: Linear Algebra</i>		
(seit WS22/23)		
Modulverantwortliche/r: Prof. Dr. Anne Leucht		
<b>Inhalte:</b>		
<ul style="list-style-type: none"> <li>• Grundlagen: Griechisches Alphabet, Aussagenlogik, Mengenlehre, Zahlbereiche, Ungleichungen, Intervalle, Potenzrechnung, Summenzeichen und Produktzeichen, Binomischer Satz</li> <li>• Vektorrechnung, Skalarprodukt, lineare Unabhängigkeit, Basis</li> <li>• Matrizenrechnung, Determinante, Rang &amp; Inverse</li> <li>• lineare Gleichungssysteme</li> <li>• Eigenwertprobleme &amp; quadratische Formen</li> <li>• allgemeiner Funktionsbegriff, Eigenschaften von Funktionen, Umkehrfunktion, rationale Funktionen</li> <li>• Folgen und Reihen: wichtige Definitionen, arithmetische und geometrische Folgen mit Beispielen im Rahmen der Kapitalverzinsung und Abdiskontierung, arithmetische und geometrische Reihen mit Beispielen im Rahmen der Renten- und Tilgungsrechnung, Grenzwerte</li> </ul>		
<b>Lernziele/Kompetenzen:</b>		
Vermittlung von mathematischen Grundkenntnissen aus den Gebieten der linearen Algebra sowie der Folgen und Reihen. 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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
keine		keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Wirtschaftsmathematik: Lineare Algebra</b>		<b>2,00 SWS</b>
<b>Lehrformen:</b> Vorlesung		
<b>Sprache:</b> Deutsch		
<b>Angebotshäufigkeit:</b> WS, SS		
<b>Literatur:</b>		
<ul style="list-style-type: none"> <li>• Jensen, U. (2017), Mathematik für Wirtschaftswissenschaftler, de Gruyter.</li> <li>• Merz, M. und Wüthrich, M. (2013), Mathematik für Wirtschaftswissenschaftler, Vahlen (München).</li> <li>• Sydsaeter K. und Hammond, P. (2018), Mathematik für Wirtschaftswissenschaftler, Pearson (München).</li> </ul>		
<b>2. Wirtschaftsmathematik: Lineare Algebra</b>		<b>1,00 SWS</b>
<b>Lehrformen:</b> Übung		
<b>Sprache:</b> Deutsch		
<b>Angebotshäufigkeit:</b> WS, SS		

**Literatur:**

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

**Prüfung**

schriftliche Modulprüfung (Klausur) / Prüfungsdauer: 60 Minuten

**Zulassungsvoraussetzung zur Modulprüfung:**

keine

<b>Modul WiMa-B-002 Wirtschaftsmathematik: Analysis</b>		6 ECTS / 180 h
<i>Mathematics for Economics and Business: Calculus</i>		
(seit WS22/23)		
Modulverantwortliche/r: Prof. Dr. Anne Leucht		
<b>Inhalte:</b>		
<ul style="list-style-type: none"> <li>• Funktionen in einer Variablen: Funktionsbegriff, Eigenschaften, Beispiele</li> <li>• Grenzwerte und Stetigkeit von Funktionen in einer Variablen</li> <li>• Differentialrechnung für Funktionen in einer Variablen: Differenzenquotient, Differentialquotient, Ableitungsregeln, Anwendung in Approximationstheorie und Optimierung, Regel von L'Hospital</li> <li>• Funktionen mehrerer Variablen: Begriffsbildung, Beispiele, Stetigkeit, partielle Differentiation, Ableitung impliziter Funktionen, totales Differential und Anwendung in Approximationstheorie</li> <li>• Krümmungsverhalten von Funktionen, Optimierung mit und ohne Nebenbedingungen</li> <li>• Integrationsrechnung: Stammfunktionen, Darboux-Summen &amp; bestimmtes Integral, unbestimmte &amp; eigentliche Integrale, Ausblick auf Integration von Funktion in mehreren Variablen</li> </ul>		
<b>Lernziele/Kompetenzen:</b>		
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.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
Inhalte der Veranstaltung Wirtschaftsmathematik: Lineare Algebra		keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Wirtschaftsmathematik: Analysis</b> <b>Lehrformen:</b> Vorlesung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS <hr/> <b>Literatur:</b> <ul style="list-style-type: none"> <li>• Jensen, U. (2017), Mathematik für Wirtschaftswissenschaftler, de Gruyter.</li> <li>• Merz, M. und Wüthrich, M. (2013), Mathematik für Wirtschaftswissenschaftler, Vahlen (München).</li> <li>• Sydsaeter K. und Hammond, P. (2018), Mathematik für Wirtschaftswissenschaftler, Pearson (München).</li> </ul>	<b>2,00 SWS</b>
<b>2. Übung zur Wirtschaftsmathematik: Analysis</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS <hr/> <b>Literatur:</b>	<b>1,00 SWS</b>

- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>• Bosch, K. (2012), Übungs- und Arbeitsbuch Mathematik für Ökonomen, Oldenbourg (München).</li><li>• Böker, F. (2013), Mathematik für Wirtschaftswissenschaftler: das Übungsbuch, Pearson (München).</li><li>• Cramer, E. (2006), Vorkurs Mathematik: Arbeitsbuch zum Studienbeginn in den Wirtschafts- und Sozialwissenschaften, Springer (Berlin).</li><li>• Merz, M. (2013), Übungsbuch für Wirtschaftswissenschaftler, Vahlen (München).</li><li>• Opitz, O. et al. (2014), Mathematik-Übungsbuch: für das Studium der Wirtschaftswissenschaften, de Gruyter Oldenbourg (Berlin).</li><li>• Schwarze, J. (2000), Mathematik für Wirtschaftswissenschaftler / 1. Grundlagen, NWB, Verl. Neue Wirtschafts-Briefe (Herne).</li></ul> |  |
|---|--|

<b>Prüfung</b>	
----------------	--

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

<b>Modul xAI-MML-B Mathematics for Machine Learning</b>		6 ECTS / 180 h
<i>Mathematics for Machine Learning</i>		
(seit SS25) Modulverantwortliche/r: Prof. Dr. Christian Ledig		
<b>Inhalte:</b> 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.		
<b>Lernziele/Kompetenzen:</b> 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 from 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.		
<b>Sonstige Informationen:</b> The lecture is conducted in English. The workload of this module is expected to be roughly as follows: <ul style="list-style-type: none"> <li>• Lecture: 22.5h (equals the 2 SWS)</li> <li>• Preparation of lectures and analysis of further sources: 30h (over the 15 weeks term)</li> <li>• Exercise classes accompanying lecture: 22.5h (equals the 2 SWS)</li> <li>• Work on the actual assignments: 75h (over the 15 weeks term)</li> <li>• Preparation for exam: 30h</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> none		
<b>Empfohlene Vorkenntnisse:</b> No specific prior knowledge is required.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Mathematics for Machine Learning</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Christian Ledig <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		<b>2,00 SWS</b>

<p><b>Lernziele:</b> c.f. module description</p> <hr/> <p><b>Inhalte:</b> The lecture will be held in English. The following is a selection of topics that will be addressed in the course</p> <ul style="list-style-type: none"> <li>• Linear Algebra (e.g., vector spaces, span, basis, rank)</li> <li>• Analytic Geometry (e.g., norms, inner product, projections)</li> <li>• Matrix decompositions (e.g., Eigenvectors, SVD)</li> <li>• Vector calculus (e.g., derivatives, Taylor series)</li> <li>• Information Theory (e.g., entropy, KL divergence)</li> <li>• Probability theory and distributions</li> <li>• Statistics (e.g., estimators, tests)</li> <li>• Optimization (e.g., gradient based)</li> <li>• Machine Learning Problems (e.g., Density estimation, Dimensionality Reduction)</li> </ul> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Marc. Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong: Mathematics for Machine Learning, Cambridge University Press, 2020</li> </ul> <p>Further literature will be announced at the beginning of the course.</p>	
<p><b>2. Mathematics for Machine Learning</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> N.N.</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b> see module description</p> <hr/> <p><b>Inhalte:</b> Further exploration of concepts discussed in the lecture by specific assignments and some programming exercises implemented predominantly in Python.</p> <hr/> <p><b>Literatur:</b> see lecture description</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> 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.</p>	

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A1 Fachstudium Mathematische Grundlagen</b>			<b>21</b>		
EESYS-SaD-B	Statistik und Data Science	SS, jährlich(1)	9	3 Vorlesung 3 Übung	schriftliche Prüfung (Klausur) 90 Minuten
WiMa-B-001	Wirtschaftsmathematik: Lineare Algebra	WS, SS(1)	6	2 Vorlesung 1 Übung	schriftliche Modulprüfung (Klausur) 60 Minuten
WiMa-B-002	Wirtschaftsmathematik: Analysis	WS, SS(1)	6	2 Vorlesung 1 Übung	schriftliche Modulprüfung (Klausur) 60 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A2 Fachstudium Informatik</b>			<b>51 - 63</b>		
<b>Pflichtbereich: A2</b>			<b>51</b>		
AI-AuD-B	Algorithmen und Datenstrukturen	SS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 105 Minuten
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 2 Übung	schriftliche Prüfung (E-Prüfung) 180 Minuten
Inf-GRABS-B	Grundlagen der Rechnerarchitektur und Betriebssysteme	SS, jährlich(1)	9	4 Vorlesung 2 Praktikum, Übung	schriftliche Prüfung (E-Prüfung) 90 Minuten 90 Minuten
MOBI-DBS-B	Datenbanksysteme	WS, SS	6	4 Vorlesung, Übung	schriftliche Prüfung (E-Prüfung) 105 Minuten
PSI-IntroSP-B	Introduction to Security and Privacy	WS, jährlich(1)	6	2 Vorlesung 2 Übung	Testat 90 Minuten schriftliche Prüfung (E-Prüfung) 120 Minuten
SWT-FSE-B	Foundations of Software Engineering	SS, jährlich	6	3 Vorlesung 3 Übung	schriftliche Prüfung (Klausur) 120 Minuten
<b>Wahlpflichtbereich: A2</b>			<b>0 - 12</b>		
AlgoK-AK-B	Algorithmen und Komplexität	SS, jährlich(1)	6	2 Vorlesung und Übung	mündliche Prüfung
AlgoK-ALDAI-B	Algorithms and logic in data science and AI	WS, jährlich(WS25/26)	6	2 Vorlesung und Übung	schriftliche Prüfung (Klausur) 90 Minuten
COMNET-RN-B	Rechnernetze	WS, jährlich	6	4 Vorlesung und Übung	Sonstiges
DT-CPP-B	Einführung in die Systemprogrammierung in C++	WS, jährlich(1)	6	4 Vorlesung und Übung	Portfolio 4 Monate
ESE-SRE-B	Software Requirements Engineering	SS, jährlich	6	2 Vorlesung 2 Übung	Sonstiges

## Modultabelle

GAMES-Java-B	Objektorientierte Programmierung mit Java	WS, jährlich(2025/2026)	3	2 Vorlesung und Übung	schriftliche Prüfung (Klausur) 90 Minuten
Gdl-MTL-B	Modal and Temporal Logic	WS, jährlich	6	4 Vorlesung und Übung	mündliche Prüfung
Inf-LBR-B	Logik und Berechenbarkeit	SS, jährlich(SS 2025)	9	6 Vorlesung und Übung	schriftliche Prüfung (Klausur) 135 Minuten
Inf-Prog-C-B	Einführung in die C-Programmierung	SS, jährlich(1)	3	1 Vorlesung 1 Praktikum, Übung	schriftliche Prüfung (E-Prüfung) 90 Minuten
MOBI-DE-B	Data Engineering	SS, jährlich(1)	6	4 Vorlesung und Übung	Hausarbeit mit Kolloquium (MOBI-DE-B) 2 Monate
SYSNAP-SNAP-B	Systemnahe Programmierung	WS, jährlich(W 2026/27)	6	4 Vorlesung und Übung	Portfolio 3 Monate 3 Monate

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A3 Fachstudium Angewandte Informatik</b>			<b>39 - 51</b>		
<b>Pflichtbereich: A3</b>			<b>9</b>		
AI-Einf-B	Einführung in die Angewandte Informatik	WS, jährlich(1)	3	2 Vorlesung und Übung	schriftliche Prüfung (Klausur) 60 Minuten
KogSys-KI-B	Einführung in die Künstliche Intelligenz	SS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Modulprüfung (Klausur) 105 Minuten
<b>Wahlpflichtbereich: A3</b>			<b>30 - 42</b>		
AI-WebT-B	Web-Technologien	SS, jährlich(SS)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
AISE-DO-B	DevOps für KI-Systeme	WS, jährlich(1)	6	2 Übung 2 Vorlesung	schriftliche Prüfung (Klausur) 90 Minuten
AISE-LKR-B	Logische Wissensrepräsentation und Schließen	WS, jährlich(1)	6	2 Übung 2 Vorlesung	schriftliche Prüfung (Klausur) 90 Minuten
CG-CGA-B	Computergrafik und Animation	WS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
DS-IDS-B	Einführung in die Dialogsysteme	WS, jährlich(1)	6	2 Vorlesung 2 Übung	Sonstiges 30 Minuten
HCI-IS-B	Interaktive Systeme	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten mündliche Prüfung
HCI-KS-B	Kooperative Systeme	SS, jährlich	6	2 Vorlesung 2 Übung	mündliche Prüfung schriftliche Prüfung (Klausur) 90 Minuten
HCI-US-B	Ubiquitäre Systeme	WS, jährlich	6	2 Vorlesung 2 Übung	mündliche Prüfung schriftliche Prüfung (Klausur) 90 Minuten
KogSys-ML-B	Einführung in Maschinelles Lernen	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 105 Minuten

## Modultabelle

MI-EMI-B	Einführung in die Medieninformatik	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 105 Minuten
MII-ROB-B	Einführung in die Robotik	WS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
NLProc-ALV-B	Algorithmisches Sprachverstehen	WS, jährlich(1)	6	4 Vorlesung und Übung	schriftliche Prüfung (Klausur) 60 Minuten
NLProc-IRTM-B	Information Retrieval and Text Mining	SS, jährlich(1)	6	4 Vorlesung und Übung	schriftliche Prüfung (Klausur) 60 Minuten
UxD-G-M	Grundlagen des Gestaltens	SS, jährlich(1)	6	2 Vorlesung 2 Übung	Portfolio 140 Stunden
VIS-GIV-B	Grundlagen der Informationsvisualisierung	SS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
xAI-MML-B	Mathematics for Machine Learning	SS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
	<b>A4 Anwendungskontext Angewandte Informatik</b>		<b>18 - 30</b>		

In der Modulgruppe A4 sind Module im Umfang von insgesamt 18 bis 30 ECTS#Punkten zu absolvieren. Die Module dürfen nicht aus dem Modulangebot der Fakultät WIAI stammen oder diesem Modulangebot gleichwertig sein. Es können Module eines oder mehrerer anderer Fächer studiert werden. Beim Studium mehrerer Fächer müssen in zwei verschiedenen Fächern jeweils mindestens 9 ECTS-Punkte erbracht werden.

Module wirtschaftswissenschaftlicher Fächer können im Umfang von bis zu 18 ECTS#Punkten eingebracht werden.

Es sind unter Berücksichtigung der oben genannten Einschränkungen beispielsweise Module aus dem Nebenfachangebot der 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. Ergibt sich aufgrund der Modulvorgaben des Anwendungsfachs ein Mindestvolumen, welches die genannte Minimalanforderung von 9 ECTS-Punkten überschreitet, z.B. 15 ECTS, so ist diese Vorgabe zu berücksichtigen. Die von der anbietenden Fakultät in der Studien- und Fachprüfungsordnung und im jeweiligen Modulhandbuch festgelegten Regelungen zur Anmeldung und Anrechnung von Modulteilprüfungen einzelner Module und ihrer zugeordneten Lehrveranstaltungen sind bindend. Im Hinblick auf einzelne Anwendungsfächer gelten 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.unibamberg.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.

Ferner ist das Anwendungsfach "Statistik und Ökonometrie" mit den unten gelisteten Modulen einbringbar. Dieses wird nicht den Wirtschaftswissenschaften zugerechnet.

Für Module der Universität Bamberg aus dem Fach Psychologie gilt zudem Folgendes:

- Zusätzlich zum Pflichtmodul „Einführung in die Psychologie, ihre Geschichte und ethische Grundlagen für (Angewandte) Informatik und IRD“ können ein bis zwei Wahlpflichtmodule im Umfang von je 6 ECTS#Punkten eingebracht werden.
- Es stehen folgende Wahlpflichtmodule zur Auswahl:
  1. Arbeits- und Organisationspsychologie für (Angewandte) Informatik und IRD
  2. Allgemeine Psychologie I für (Angewandte) Informatik und IRD
  3. Allgemeine Psychologie II für (Angewandte) Informatik und IRD
  4. Biologische Psychologie für (Angewandte) Informatik und IRD
  5. Persönlichkeitspsychologie für (Angewandte) Informatik und IRD

## Modultabelle

### 6. Sozialpsychologie für (Angewandte) Informatik und IRD.

- Die Modulprüfung wird jeweils durch schriftliche Prüfung (90 Minuten) oder mündliche Prüfung (15 Minuten) erbracht.
- Das konkrete Angebot der aus dem Fach Psychologie wählbaren Module, sowie die konkreten Modulbeschreibungen sind dem Modulhandbuch für Module des Fachs Psychologie, die im Rahmen des Bachelor- und des Masterstudiengangs Angewandte Informatik, des Bachelorstudiengangs Informatik sowie des Masterstudiengangs Interaction Research & Design erbracht werden können, zu entnehmen.

### Wahlpflichtbereich: Anwendungsfach Statistik und Ökonometrie 0 - 30

Die Module Statistische Programmierung mit R und Advanced Data Analysis with R unterliegen dem Vorbehalt von freien Kapazitäten im CIP-Pool.

SuStat-013-M	Introduction to Econometrics	WS, jährlich	6	4 Vorlesung, Übung	schriftliche Prüfung (Klausur) 90 Minuten
SuStat-014-M	Advanced Econometrics	SS, jährlich	6	4 Vorlesung, Übung	schriftliche Prüfung (Klausur) 60 Minuten
SuStat-071-M	Advanced Data Analysis With R	SS, jährlich	6	2 Übung	schriftliche Hausarbeit 8 Wochen schriftliche Prüfung (Klausur) 90 Minuten
SuStat-075-M	Statistische Programmierung mit R	WS, SS(1)	6	2 Übung	schriftliche Hausarbeit 8 Wochen schriftliche Modulprüfung (Klausur) 90 Minuten
SuStat-076-M	Fortgeschrittene Statistik	WS, jährlich	6	3 Vorlesung, Übung	schriftliche Prüfung (Klausur) 60 Minuten
SuStat-079-M	Analyse hochdimensionaler Daten	SS, jährlich	6	3 Vorlesung, Übung	schriftliche Prüfung (Klausur) 60 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A5 Überfachliche Qualifikationen</b>			<b>7 - 13</b>		
<b>Pflichtbereich: A5</b>			<b>7</b>		
Inf-Ment-B	Studieneinstiegsmentoring für Bachelorstudierende	WS, SS	1	0 Vorlesung 0 Kleingruppenprojekt	Portfolio 4 Monate
MI-WAIAI-B	Einführung in das wissenschaftliche Arbeiten für Informatik und Angewandte Informatik	SS, jährlich(1)	3	2 Vorlesung und Übung	Portfolio 4 Monate
PSI-EDS-B	Ethics for the Digital Society	WS, jährlich	3	2 Vorlesung	schriftliche Prüfung (E-Prüfung) 80 Minuten
<b>Wahlpflichtbereich: A5</b>			<b>0 - 6</b>		
<p>Im Wahlpflichtbereich kann frei aus den Bereichen Fremdsprachen, Philosophie/Ethik und Allgemeine Schlüsselqualifikationen gewählt werden.</p> <ul style="list-style-type: none"> <li>• Im Bereich Fremdsprachen können Module gemäß dem Angebot des Sprachenzentrums Bamberg absolviert werden. Ausgenommen sind Module in der Sprache, in der die Hochschulzugangsberechtigung erworben wurde. Ausgenommen sind darüber hinaus Module zu Fremdsprachen bis zu dem Niveau, das in der Hochschulzugangsberechtigung ausgewiesen wurde. Dies gilt nicht für die Module der Bereiche der Wirtschaftsfremdsprachen und des Bereichs IT-English. Die Module der Bereiche Deutsch als Fremdsprache und Wirtschaftsdeutsch sind ab der Niveaustufe C1 wählbar, sofern die Sprache, in der Hochschulzugangsberechtigung erworben wurde, nicht Deutsch war. Einzelheiten, insbesondere die zur Auswahl stehenden Module sowie die jeweils abzulegenden Modulprüfungen und Modulteilprüfungen, sind in der Prüfungsordnung und dem Modulhandbuch für sprachpraktische Module der Otto-Friedrich-Universität Bamberg festgelegt.</li> <li>• Im Bereich Philosophie/Ethik sind auf Antrag Module wählbar, die der Ethik oder der Philosophie zuzuordnen sind und im Studium Generale angeboten werden.</li> <li>• Im Bereich Allgemeine Schlüsselqualifikationen stehen neben Modulen des Zentrums für Schlüsselkompetenzen gemäß Studien- und Fachprüfungsordnung für Module und Zertifikate im Bereich der Schlüsselkompetenzen § 4 Abs 2 folgende Module zur Auswahl:</li> </ul>					
AIC-IITP-B	Internationales IT Projektmanagement	SS, jährlich(1)	6	4 Vorlesung und Übung	schriftliche Prüfung (Klausur) 90 Minuten
HCI-DISTP-B	Design Interaktiver Systeme: Theorie und Praxis	SS, jährlich(1)	6	1 Vorlesung und Übung 1 Übung	Kolloquium 30 Minuten Kolloquium 30 Minuten

## Modultabelle

---

KogSys-GAI-B	Genderaspekte in der Informatik	SS, jährlich(1)	3	2 Seminar	Hausarbeit mit Referat 4 Monate 30 Minuten
PSI-DatSchu-B	Datenschutz	SS, jährlich(1)	3	2 Vorlesung	schriftliche Prüfung (Klausur) 90 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A6 Seminare und Projekte</b>			<b>18 - 24</b>		
<b>Wahlpflichtbereich: Seminare</b>			<b>6</b>		
<b>Fach: Seminare der Angewandten Informatik</b>			<b>3 - 6</b>		
AI-Seminar1-B	Bachelorseminar 1 der Fächergruppe Angewandte Informatik	WS, SS(1)	3	2 Seminar	Hausarbeit mit Referat
AI-Seminar2-B	Bachelorseminar 2 der Fächergruppe Angewandte Informatik	WS, SS(1)	3	2 Seminar	Hausarbeit mit Referat
<b>Fach: Seminare der Informatik oder Wirtschaftsinformatik</b>			<b>0 - 3</b>		
Inf-Seminar1-B	Bachelorseminar 1 der Fächergruppe Informatik	WS, SS	3	2 Seminar	Hausarbeit mit Referat
WI-Seminar-B	Bachelorseminar aus der Fächergruppe Wirtschaftsinformatik	WS, SS	3	2 Seminar	Hausarbeit mit Referat
<b>Wahlpflichtbereich: Projekte</b>			<b>12 - 18</b>		
<b>Fach: Projekte der Angewandten Informatik</b>			<b>12 - 18</b>		
AI-Projekt1-B	Bachelorprojekt 1 der Fächergruppe Angewandte Informatik	WS, SS(1)	6	4 Projektseminar	Hausarbeit mit Kolloquium
AI-Projekt2-B	Bachelorprojekt 2 der Fächergruppe Angewandte Informatik	WS, SS(1)	6	4 Projektseminar	Hausarbeit mit Kolloquium
AI-Projekt3-B	Bachelorprojekt 3 der Fächergruppe Angewandte Informatik	WS, SS(1)	6	4 Projektseminar	Hausarbeit mit Kolloquium
<b>Fach: Projekte der Informatik oder Wirtschaftsinformatik</b>			<b>0 - 6</b>		
ESE-SEL-B	Software Engineering Lab	keine Angabe	6	4	Hausarbeit mit Kolloquium
Inf-Projekt1-B	Bachelorprojekt 1 der Fächergruppe Informatik	WS, SS	6	4 Projektseminar	Hausarbeit mit Kolloquium
WI-Projekt-B	Bachelorprojekt aus der Fächergruppe Wirtschaftsinformatik	WS, SS	6	4 Übung	Hausarbeit mit Kolloquium

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
	<b>A7 Bachelorarbeit</b>		<b>12</b>		
AI-Thesis-B	Bachelorarbeit im Studiengang Angewandte Informatik	WS, SS(1)	12		Hausarbeit mit Kolloquium 4 Monate