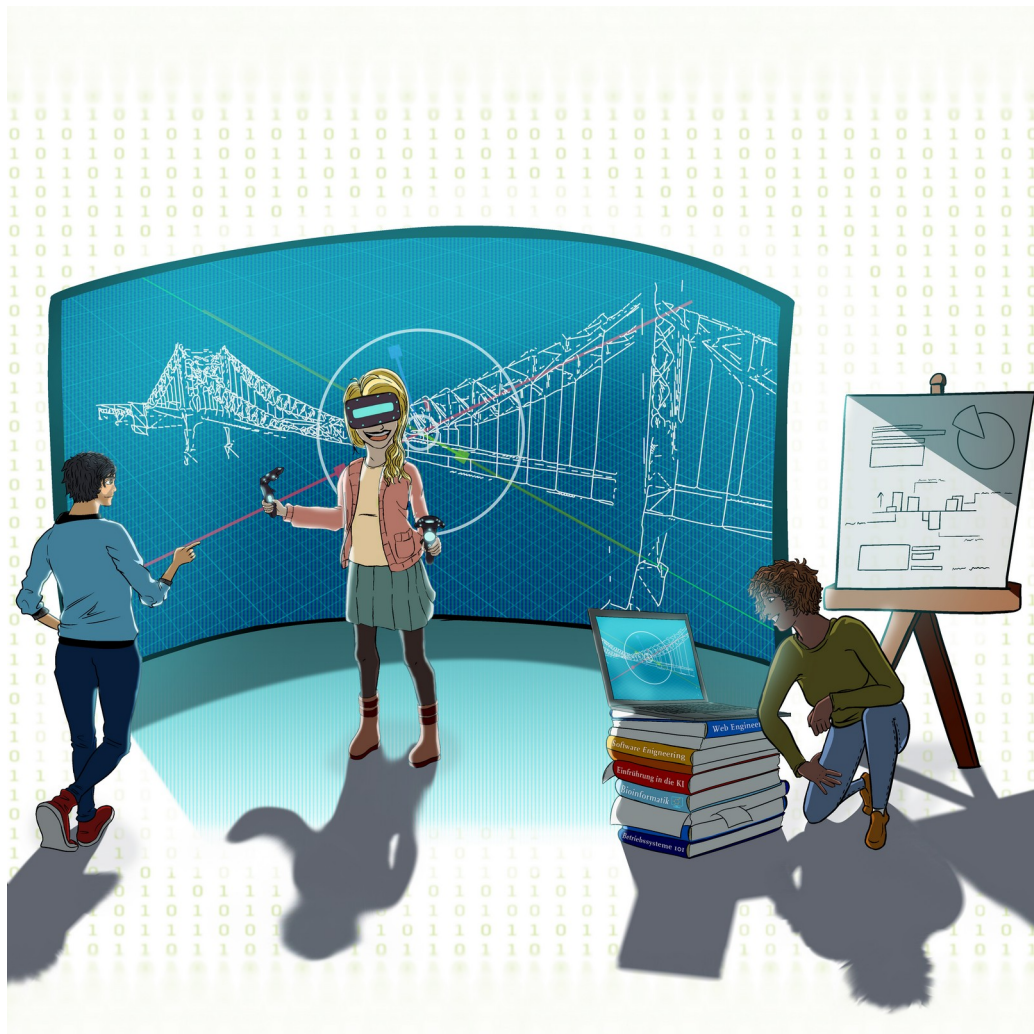


Modulhandbuch Bachelor of Science (B.Sc.)

Angewandte Informatik [PO22]

Stand: Wintersemester 2025/2026

<https://informatik.rub.de/studium/studiengaenge/ai/bsc/>



Studienplan Bachelor Angewandte Informatik PO 22

Nr	Modul	Umfang bzw. Mind. Umfang Modul (CP)	Empfohlenes Semester	Bewertung
Pflichtbereich				
1	Höhere Mathematik 1	9	1	benotet
2	Informatik I	12	1	benotet
3	Technische Informatik 1	5	1	benotet
4	Höhere Mathematik 2	9	2	benotet
5	Logik	5	2	benotet
6	Informatik 2	8	2	benotet
7	Computernetze	5	2	benotet
8	Programmierung und Programmiersprachen	6	2	benotet
9	Statistik und Wahrscheinlichkeitsrechnung	5	3	benotet
10	Informatik 3	8	3	benotet
11	Software-Engineering	5	3	benotet
12	Projektmanagement	5	3	benotet
13	Wirtschaftlichkeitsanalyse	5	3	benotet
14	Einführung in die künstliche Intelligenz	5	4	benotet
15	Datenbanksysteme	7	4	benotet
16	Web-Engineering	5	4	benotet
17	Betriebssysteme	5	4	benotet
18	Software Engineering Praktikum	5	4	benotet
19	Datenschutz	5	5	benotet
Wahlpflichtbereich				
20	Vertiefungspraktikum*	3	4	unbenotet
22	Vertiefungsseminar*	3	5	benotet
21	Studienprojekt**	8	5	benotet
23	Vertiefungsmodule***	23	5-6	benotet
Wahlbereich				
24	Freie Wahlmodule	9	1-6	unbenotet
Bachelorarbeit				
25	Abschlussarbeit	12+3	6	benotet
Summe:		180		

* Informationen zu den im Semester wählbaren Vertiefungspraktika und Vertiefungsseminaren befinden sich im Vorlesungsverzeichnis.

** Die wählbaren Studienprojekte werden jeweils zum Ende eines Semesters fürs Folgesemester vorgestellt.

*** Hier müssen Vertiefungsmodule aus Anwendungsbereichen der Informatik im Umfang von mindestens 23 LP gewählt werden. Informationen zu den wählbaren Modulen befinden sich im jeweils aktuellen Modulhandbuch.

**** Hier können (nahezu) alle Veranstaltungen des Vorlesungsverzeichnisses der RUB, sowie Veranstaltungen im Rahmen der Universitätsallianz Ruhr gewählt werden.

Angebote Vertiefungsmodule

Lehrveranstaltung	Lehrinheit	Umfang (CP)	Semester	Bewertung
Vertiefungsmodule				
Ingenieurinformatik				
Geometrische Modellierung und Visualisierung	Baulng	6	WS	benotet
Grundlagen der Automatisierungstechnik	MB	5	WS	benotet
Numerische Mathematik	Mathe	5	WS	benotet
Virtuelle Produktmodellierung und -visualisierung	MB	5	WS	benotet
Digitaltechnik	Informatik	5	SS	benotet
Fertigungsautomatisierung	MB	5	SS	benotet
Menschenzentrierte Robotik	MB	6	SS	benotet
Bioinformatik				
Bioinformatics for Molecular and Cellular Structures (ehemals Grundlagen der Bioinformatik)	Biologie	5	WS	benotet
Methoden der Bioinformatik	Biologie	5	SS	benotet
Computerlinguistik				
Einführung in die Linguistik	Philologie	8	WS	benotet
Einführung in die Computerlinguistik	Philologie	6	SS (ab SS 26)	benotet
Computerlinguistische Programmierung	Philologie	9	SS (ab SS 26)	benotet
Natural Language Processing with Deep Learning	Informatik	5	WS	benotet
Anwendungen der Computerlinguistik	Philologie	5 bzw. 8	Letztmalig SS 24	benotet
Methoden der Computerlinguistik	Philologie	5	Letztmalig WS 24/25	benotet
Kryptographie und Theoretische Informatik				
Einführung in die Kryptographie I	Informatik	5	WS	benotet
Einführung in die Kryptographie II	Informatik	5	SS	benotet
Highlights of Theoretical Computer Science	Informatik	10	SS	benotet
Kryptographie auf hardwarebasierten Plattformen	Informatik	5	Letztmalig WS 22/23	benotet
Neuroinformatik				
Artificial Neural Networks	Informatik	6	WS	benotet
Introduction to Data Science	Informatik	5	WS	benotet
Introduction to Computational Neuroscience	Informatik	6	SS	benotet
Introduction to Neural Data Science	Informatik	6	Letztmalig WS 23/24	benotet
Programmier- und Simulationstechnik				
Algorithm Paradigms (ehemals Algorithmenparadigmen)	Informatik	5	WS	benotet
Programming for Modern Machine Learning	Informatik	6	WS/SS	benotet
System Performance Evaluation	Informatik	5	WS	benotet
Technische Informatik 3 - Hardware Programmierung	Informatik	5	WS	benotet
Functional Programming	Informatik	5	WS	benotet
Human-Computer Interaction	Informatik	6	WS	benotet
Distributed Systems Algorithms (alter Name Distributed Systems)	Informatik	5	SS	benotet
Mathematics for Modelling and Data Analysis	Informatik	5	SS	benotet
Mensch-Maschine-Interaktion	AW	5	Ausgelaufen SS 25	benotet
Game Development	Informatik	6	Letztmalig SS 23	benotet
Nebenläufige Programmierung	Informatik	5	Letztmalig SS 23	benotet
IT in Wirtschaft, Management und Gesellschaft				
Geschäftsprozess-Management	WiWi	5	WS	benotet
Wirtschaft und Digitalisierung	WiWi	5	WS	benotet
Inklusives IT-Design	AW	6	SS (nicht im SS 25)	benotet
Agent-based Modeling in Economics and Business	WIWI	5	Ausgelaufen SS 25	benotet

Angebote Vertiefungsseminare und Vertiefungspraktika

	Lehrveranstaltung	Lehreinheit	Umfang (CP)	Semester	Bewertung
Vertiefungsseminare					
	Seminar Computing in Engineering (ehemals Seminar Ingenieurinformatik)	Baulng	3	WS/SS	benotet
	Seminar Bioinformatik	Biologie / MPC	3	WS/SS	benotet
	Seminar Security Engineering	Informatik	3	WS/SS	benotet
	Seminar Safety and Reliability Artificial Intelligence	Informatik	3	WS/SS	benotet
	Seminar Automated Software Engineering	Informatik	3	WS/SS	benotet
	Seminar Ressourceneffiziente Systemsoftware	Informatik	3	WS/SS	benotet
	Seminar Modern Programming Languages	Informatik	3	WS	benotet
	Seminar Mobile Network Security	Informatik	3	WS	benotet
	Perlen der theoretischen Informatik (ehemals Grenzen in der theoretischen Informatik)	Informatik	3	WS	benotet
	Seminar Distributed Systems	Informatik	3	WS	benotet
	Seminar Algorithmen	Informatik	3	WS	benotet
	Seminar: Introduction to Bayesian Modeling	Informatik	3	WS	benotet
	Proseminar - Die KI-Verordnung der EU	Jura	3	WS	benotet
	Seminar Perlen der Logik (ehemals Satisfiability)	Informatik	3	SS	benotet
	Seminar Computational Neuroscience	Informatik	3	SS	benotet
	Seminar From Biological to Artificial Neural Networks	Informatik	3	SS	benotet
	Seminar Networked Systems	Informatik	3	SS	benotet
	Seminar Reinforcement Learning	Informatik	3	SS	benotet
	Seminar on Applied Privacy and Anonymity	Informatik	3	SS	benotet
	Seminar Methaphern, Korpus- und Computerlinguistische Ansätze	Philologie	3	SS	benotet
	Seminar Implementation Security	Informatik	3	Letztmalig SS 23	benotet
	Seminar Symmetrische Kryptanalyse	Informatik	3	Letztmalig WS 22/23	benotet
	Seminar on Knowledge Graphs	Informatik	3	Letztmalig WS 22/23	benotet
	Machine Learning Applications	Informatik	3	Letztmalig WS 23/24	benotet
	Algorithms for Decision Making	Informatik	3	Letztmalig WS 23/24	benotet
	Seminar on Current Topics for Systems Security and Privacy	Informatik	3	Letztmalig WS 23/24	benotet
	Seminar CL (Im WS 24/25: Informationsstatus und Koreferenzauflösung (050042))	Philologie	3	Letztmalig WS 24/25	benotet
	Seminar Approximationsalgorithmen	Informatik	3	Letztmalig SS 24	benotet
Vertiefungspraktika					
	Grundlagen der Roboterprogrammierung	MB	3	WS/SS	unbenotet
	Python programming and basic machine learning	Informatik	3	WS/SS	unbenotet
	An Introduction to Python for Data Analysis (ehemals Introduction to Python)	Informatik	3	SS	unbenotet
	Erklärbare Künstliche Intelligenz - Programmierpraktikum	AW	3	SS	unbenotet
	Open Neural Data	Informatik	3	SS	unbenotet
	Python in der Praxis	Philologie	3	SS (ab SS 26)	unbenotet
	Advanced Python Programming	Informatik	3	WS	unbenotet
	Lab Course: Challenging Problems in Reinforcement Learning	Informatik	3	WS	unbenotet
	Lab course: Introduction to Bayesian Modeling	Informatik	3	WS	unbenotet
	Autonomous Driving Simulation Lab	Informatik	3	WS (nicht im WS 25/26)	unbenotet
	Praktikum Systemsoftwaretechnik	Informatik	3	WS	unbenotet
	Praktikum Anlagenprogrammierung	MB	3	WS	unbenotet
	Praktikum Rust	Informatik	3	Letztmalig WS 23/24	unbenotet

Unreal C++ Praktikum	Informatik	3	WS (letztmalig WS 24/25)	unbenotet
----------------------	------------	---	--------------------------	-----------

Abkürzungen:

AW: Institut für Arbeitswissenschaft

BauIng: Fakultät für Bau- und Umweltingenieurwissenschaften

ETIT: Fakultät für Elektrotechnik und Informationstechnik

MB: Fakultät für Maschinenbau

WIWI: Fakultät für Wirtschaftswissenschaft

MPC: Medizinisches Proteom Center

Jura: Juristische Fakultät

SS: Sommersemester

WS: Wintersemester

CP: Creditpoints

MODULHANDBUCH

Übersicht der Module

Angewandte Informatik - Bachelor (1-Fach, PO 2022)

Pflichtbereich

Höhere Mathematik 1
Wirtschaftlichkeitsanalyse
Informatik 1
Technische Informatik 1
Höhere Mathematik 2
Informatik 2
Computernetze
Programmierung und Programmiersprachen
Logik
Statistik und Wahrscheinlichkeitsrechnung
Informatik 3
Software Engineering
Projektmanagement
Einführung in die künstliche Intelligenz
Datenbanksysteme
Web-Engineering
Betriebssysteme
Software Engineering Praktikum
Datenschutz

Wahlpflichtbereich

Algorithm Paradigms
Bioinformatics for Molecular and Cellular Structures
Computerlinguistische Programmierung
Digitaltechnik
Distributed Systems Algorithms
Einführung in die Computerlinguistik
Einführung in die Kryptographie 1
Einführung in die Kryptographie 2
Einführung in die Linguistik
Fertigungsautomatisierung
Functional Programming

Geometrische Modellierung und Visualisierung
Geschäftsprozess-Management
Grundlagen der Automatisierungstechnik
Highlights of Theoretical Computer Science [B.Sc.]
Human-Computer Interaction [B.Sc.]
Inklusives IT-Design (Kein Angebot im SS 25)
Introduction to Computational Neuroscience
Introduction to Data Science
Künstliche Neuronale Netze
Mathematics for Modeling and Data Analysis
Menschenzentrierte Robotik
Methoden der Bioinformatik
Natural language processing with deep learning [B.Sc.]
Numerische Mathematik
Programming for Modern Machine Learning
System Performance Evaluation
Technische Informatik 3
Virtuelle Produktmodellierung und -visualisierung
Wirtschaft und Digitalisierung
Studienprojekt
Vertiefungspraktikum (Angewandte Informatik)
Vertiefungsseminar (Angewandte Informatik)

Wahlbereich

Freie Wahlmodule

Bachelorarbeit

Abschlussarbeit (Bachelor AI)

Titel des Moduls: Höhere Mathematik 1					
Modul-Nr./Code	Credits 9 CP	Workload 270 h	Semester 1	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Vorlesung: Höhere Mathematik 1 (150160) Übung: Höhere Mathematik 1 (150161)			Kontaktzeit 90 h	Selbststudium 180 h	Gruppengröße 120 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: PD. Dr. Daniela Kacso Lehrende: PD. Dr. Daniela Kacso					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Mathematische Schulausbildung (Gymnasiale Oberstufe) und die Teilnahme an einem Mathematik-Vorkurs					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls: Die Studierende <ul style="list-style-type: none"> • Verstehen und wenden grundlegende mathematische Begriffe und Notationen an • Nutzen und führen die vermittelten mathematischen Methoden aus • Übertragen und wenden zugehörige Lösungsverfahren auch auf praktische Probleme an 					
Inhalt <ul style="list-style-type: none"> • Aussagenlogik, • Mengen und Abbildungen • Reelle Zahlen und algebraische Strukturen • Komplexe Zahlen • Folgen und Reihen • Stetige Funktionen • Differenzialrechnung (in \mathbb{R}) • Integralrechnung (in \mathbb{R}) 					
Lehrformen Vortrag der Lehrenden in der Vorlesung (mit zum Teil digitalen Lehrformaten), Gruppenarbeit in den Übungen, Ergänzung der Bearbeitung der Hausaufgaben in Einzel- oder Gruppenarbeit durch digitale Aufgaben, Online-Tests.					
Prüfungsformen Klausur (120 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestehen der Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 9/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Wirtschaftlichkeitsanalyse

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Vorlesung mit Übung (076000) Optionales Kolloquium (076001)			Kontaktzeit 45 h	Selbststudium 105 h	Gruppengröße 120 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Dr. Barbara Wischermann
Lehrende: Dr. Barbara Wischermann
Dr. Roland Düsing

Verwendung des Moduls

B.Sc. Angewandte Informatik

Vorkenntnisse

Keine

Lernziele (learning outcomes)

Nach dem erfolgreichen Abschluss des Moduls:

- sollen die Studierenden Grundbegriffe der Wirtschaftlichkeitsanalyse kennen
- sollen die Studierenden die verschiedenen Teilgebiete der Wirtschaftlichkeitsanalyse auseinanderhalten können
- sollen die Studierenden Aufgaben der Kostenarten-, Kostenstellen- und Kostenträgerrechnung lösen können
- sollen die Studierenden Erfolgsgrößen errechnen und beurteilen können
- sollen die Studierenden Methoden der Investitionsrechnung sicher anwenden
- sollen die Studierenden kompetent mit wirtschaftlichen Fachbegriffen umgehen können
- sollen die Studierenden in der Lage sein, sich mit Kaufleuten inhaltlich über die entsprechenden Themen auszutauschen
- sollen die Studierenden bei Budgetverantwortung und eigenen Projekten die Grundbegriffe der Wirtschaftlichkeit kennen und berücksichtigen können

Inhalt

Angewandte Informatiker werden sich im Rahmen ihrer Berufstätigkeit als Budgetverantwortliche oder im Rahmen eines Projektmanagements auch regelmäßig mit Frage der Wirtschaftlichkeit ihres Handelns auseinandersetzen haben. Darüber hinaus werden im Kundengespräch und bei der Auftragsabwicklung Kenntnisse von Wirtschaftlichkeitsgrößen und Vorteilhaftigkeitsrechnungen als Verkaufsargumente notwendig sein. Nicht zuletzt wird sich für viele Informatiker sowohl im Rahmen eigener Investitionsüberlegungen als auch bei der Entwicklung von Software-Lösungen für Kunden die Frage nach der Wirtschaftlichkeit von Investitionsalternativen stellen. Die Lehrveranstaltung „Wirtschaftlichkeitsanalyse“ wird die Studierenden mit den Grundlagen der Wirtschaftlichkeitsanalyse vertraut machen und ihnen Methoden vermitteln, mit denen sie derartige Fragestellungen beantworten können.

Gliederung:

1. Anwendung der Wirtschaftlichkeitsanalyse (u.a. Budgetverantwortung u. Projektmanagement, Kundengespräch und Auftragsabwicklung, Investitionsentscheidungen)
2. Grundbegriffe und begriffliche Abgrenzung
3. Die Wirtschaftlichkeit in der Leistungserstellung (u.a. Kostenarten-, Kostenstellen-, Kostenträgerrechnung; Erlöse; Erfolgsermittlung)

4. Verfahren der Wirtschaftlichkeitsrechnung in der Investitionsrechnung (statische und dynamische Verfahren)

Lehrformen

Vorlesung mit integrierter Übung

Prüfungsformen

Klausur (60 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Informatik 1					
Modul-Nr./Code	Credits 12 CP	Workload 360 h	Semester 1	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Vorlesung und Übung: Informatik 1 (212004) Python-Praktikum (212400)			Kontaktzeit 6 Semesterwochenstunden + 10 Tage (80h)	Selbststudium 190 h	Gruppengröße 400 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tobias Glasmachers Lehrende: Prof. Dr. Tobias Glasmachers					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik					
Vorkenntnisse Keine					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen die Teilnehmer die wichtigsten Konzepte imperativer und objektorientierter Programmierung, • können die Teilnehmer eigene Programme entwerfen und implementieren, • können die Teilnehmer mit Grundbegriffen der Informatik wie etwa Korrektheit, Laufzeit, Boole'scher Algebra, Invarianten und abstrakten Datentypen arbeiten, • können die Teilnehmer die einfache Datenstrukturen (Arrays, Dictionaries) gezielt einsetzen und kennen Standardalgorithmen darauf, insbesondere zum Sortieren von Arrays. 					
Inhalt <ul style="list-style-type: none"> • Imperative Programmierung (Variablen, Kontrollstrukturen, Funktionen und Rekursion, Fehlerbehandlung, Ereignisbehandlung) • einfache Datenstrukturen (Array und Dictionary) • Objektorientierung (Klassen, Sichtbarkeit, Schnittstellen, Vererbung) • Einführung in eine Reihe von Informatik-Konzepten (Invarianten, Laufzeitanalyse, Sortieralgorithmen, Repräsentation von Daten im Rechner, Boole'sche Algebra) 					
Lehrformen Vorlesung mit Übung plus zweiwöchiges Blockpraktikum Die Vorlesung nutzt das Flipped-Classroom Lehrformat. Sämtliches Vorlesungsmaterial steht online zur Verfügung.					
Prüfungsformen Klausur (150 Minuten) Erfolgreiche Teilnahme am Praktikum					
Voraussetzungen für die Vergabe von Credits Teilnahme am Python-Praktikum und bestandene Klausur.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 12/168: B.Sc. Angewandte Informatik 12/158: B.Sc. Informatik					

Titel des Moduls: Technische Informatik 1 Technical Computer Science 1					
Modul-Nr./Code	Credits 5 CP	Workload 150 Stunden	Semester siehe Prüfungsordnung / see Examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Technische Informatik 1 - Rechnerarchitektur (212013)			Kontaktzeit 4 SWS	Selbststudium	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Yuval Yarom Lehrende: Prof. Yuval Yarom					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik B.Sc. IT-Sicherheit/ Informationstechnik					
Vorkenntnisse Keine					
Lernziele (learning outcomes) Upon successful completion of the course, students will be able <ul style="list-style-type: none"> • Identify the main components of a modern processor, describe their functionality, and demonstrate how they each contribute to program execution • Evaluate computer system based on their design, and assess their impact on program performance • Design and develop simple programs in assembly language 					
Inhalt The course introduces the structure and function of modern computers. It explores the various components that comprise the computer, including the execution pipeline, the memory subsystem, data storage, and external devices. A main focus of the course is exploring and analyzing program execution. This start from practical experience with assembly programming and develops to in-depth analysis of how the processor interprets and performs a program. In particular, the course identifies trade-offs made in processor design and their impact on performance.					
Lehrformen Contact teaching consists of two hours of lecture per week, supplemented by two hours of practical exercise sessions. Beyond contact hours, students are expected to read textbook chapters and to answer take-home exercises.					
Prüfungsformen Klausur (120 Minuten). Bis zu 10% Bonus für Hausaufgaben.					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)					

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/150: B.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/ Informationstechnik [PO 20]

Titel des Moduls: Höhere Mathematik 2					
Modul-Nr./Code	Credits 9 CP	Workload 270 h	Semester 1	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Vorlesung (150162) Übung (150163)			Kontaktzeit 90 h	Selbststudium 180 h	Gruppengröße 150 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: PD. Dr. Daniela Kacso Lehrende: PD. Dr. Daniela Kacso					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Mathematische Schulausbildung (Gymnasiale Oberstufe) und die Teilnahme an einem Mathematik-Vorkurs					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls: Die Studierenden <ul style="list-style-type: none"> • Verstehen und wenden grundlegende mathematische Begriffe und Notationen an • Sind in der Lage die Themen zu erläutern und Einsatzmöglichkeiten zu benennen • Können Formalismen und Verfahren auswählen und ausführen sowie die erzielten Ergebnisse interpretieren • Übertragen und wenden zugehörige Lösungsverfahren auch auf praktische Probleme an 					
Inhalt <ul style="list-style-type: none"> • Potenzreihen und Fourierreihen • Vektorräume • Lineare Gleichungssysteme • Matrizen • Determinanten • Eigenwerte und Eigenvektoren • Differenzialrechnung (in \mathbb{R}^n) • Gewöhnliche Differenzialgleichungen 					
Lehrformen Vortrag der Lehrenden in der Vorlesung (mit zum Teil digitalen Lehrformaten), Gruppenarbeit in den Übungen, Ergänzung der Bearbeitung der Hausaufgaben in Einzel- oder Gruppenarbeit durch digitale Aufgaben, Online-Tests					
Prüfungsformen Klausur (120 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestehen der Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 9/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Informatik 2 Computer Science 2					
Modul-Nr./Code	Credits 8 CP	Workload 240 h	Semester 2	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Informatik 2 - Algorithmen und Datenstrukturen (211002)			Kontaktzeit 90 h	Selbststudium 150 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Maike Buchin Lehrende: Prof. Maike Buchin					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/ Informationstechnik					
Vorkenntnisse Inhalte der Module Informatik 1 und Mathematik 1 bzw. Höhere Mathematik 1, insbesondere Programmieren und lineare Algebra					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls: <ul style="list-style-type: none"> • können Studierende Algorithmen formal beschreiben und deren Korrektheit beweisen • können Studierende die Laufzeit und den Speicherbedarf von Algorithmen und Datenstrukturen analysieren und bewerten • kennen Studierende grundlegende Datenstrukturen • kennen Studierende grundlegende Schemata zum Entwurf von Algorithmen sind Studierende in der Lage, Algorithmen und Datenstrukturen für spezifische Probleme zu entwickeln • haben die Studierenden die Grundlagen der Programmiersprache Python kennengelernt 					
Inhalt Die Vorlesung gibt einen systematischen Überblick über den Entwurf und die Analyse von Algorithmen und Datenstrukturen. Dazu werden zunächst grundlegende Methoden der Analyse (insbesondere Korrektheit, Laufzeit und Speicherbedarf) von Algorithmen vorgestellt. Anschließend werden einige Algorithmen zum Sortieren und Suchen analysiert. Ebenfalls werden verschiedene grundlegende Datenstrukturen (Listen, Felder, Suchbäume und Heaps) vorgestellt. Schließlich werden Graphen betrachtet, und zwar ihre Darstellung und diverse Algorithmen auf Graphen (Durchläufe, kürzeste Wege, minimale Spannbäume). In den Übungen lernen die Studierenden sowohl die theoretische Analyse von Algorithmen und Datenstrukturen als auch deren praktische Umsetzung in eine moderne Programmiersprache (z.B. Python).					
Lehrformen Hörsaalvorlesung mit Medienunterstützung und theoretische sowie praktische Übungen am Rechner					
Prüfungsformen Klausur (150 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)					

8/158: B.Sc. Informatik [PO 22]

8/165: B.Sc. Informatik [PO 20]

8/168: B.Sc. Angewandte Informatik [PO 22]

8/170: B.Sc. Angewandte Informatik [PO 20]

8/150: B.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

8/149: B.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

Titel des Moduls: Computernetze Computer Networks					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 2	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Computernetze (211006)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 400 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Jörg Schwenk Lehrende: Prof. Dr. Jörg Schwenk					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit / Informationstechnik					
Vorkenntnisse					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende die wichtigsten Standards, die das heutige Internet verwendet. • kennen Studierende grundlegende Angriffskonzepte auf Computernetzwerke • verstehen Studierende den Zusammenhang zwischen den einzelnen Schichten eines Computernetzwerks und der darin enthaltenen Protokolle • können Studierende die wichtigsten Netzwerktools für Analysezwecke anwenden 					
Inhalt Die Vorlesung gibt eine Einführung in grundlegenden Protokollen und Anwendungen von Computernetzen. Der Schwerpunkt der Vorlesung liegt auf Standardprotokollen und -Algorithmen, wie sie in modernen Computernetzwerken (zum Beispiel im Internet) eingesetzt werden. Anhand eines Schichtenmodells werden die wichtigsten Grundlagen nach dem Top-Down Ansatz vorgestellt und analysiert. Dazu gehören zum Beispiel auf der obersten Schicht DNS und HTTPS im Application Layer; TCP und UDP im Transport Layer; IPv4/IPv6 und Routing Algorithmen im Network Layer; sowie MAC und ARP im untersten Link Layer. Neben der reinen Funktionsweise dieser Standards werden Sicherheitsaspekte auf allen Schichten betrachtet. Ergänzend zur Vorlesung werden Übungsaufgaben über die eLearning Plattform Moodle gestellt und in der Übungsstunde besprochen. Weiterhin wird in jeder Übung ein "Tool der Woche" vorgestellt. Dabei handelt es sich jeweils um eine spezielle Software, die man als "Netzwerker" unbedingt kennen sollte (z.B. traceroute, nmap, ...). Alle besprochenen Tools sind frei verfügbar und werden den Studenten als eine Lernplattform (virtuelle Maschine) zur Verfügung gestellt. Als Primärliteratur wird "Computernetzwerke: Der Top-Down Ansatz" von Kurose und Ross (Pearson Verlag) verwendet.					
Lehrformen Moodle-Unterstützte Hausaufgaben mit praxisnahen, computerunterstützten Übungen. Tool-der-Woche: Vorstellung, Einarbeitung, und Verwendung von Netzwerkrelevanten Computer analysetools.					
Prüfungsformen Klausur (120 Minuten)					

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

Titel des Moduls: Programmierung und Programmiersprachen
Programming and programming languages

Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester 2	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Programmierung und Programmiersprachen: Vorlesung und Übung (211053)			Kontaktzeit 60 h	Selbststudium 120 h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr. Nils Jansen
 Lehrende: Prof. Dr. Nils Jansen

Verwendung des Moduls

B. Sc. Informatik

 B.Sc. Angewandte Informatik

Vorkenntnisse

Inhalte des Moduls Informatik 1

Lernziele (learning outcomes)

Ziel dieses Kurses ist es, ein umfassendes Verständnis der wichtigsten Programmierparadigmen zu vermitteln: Imperative und objektorientierte Programmierung, funktionale Programmierung und logische Programmierung. Der Kurs stellt den Studierenden praktische und moderne Programmierübungen bereit, die ihnen bei der Lösung realer Programmierprobleme helfen. Der Schwerpunkt des Kurses liegt auf der Entwicklung, Analyse und Verifizierung von Code, der solche Probleme effizient löst. Darüber hinaus erlernen die Studierenden im Rahmen der Programmierübungen den Umgang mit modernen Softwareentwicklungstools wie IDEs und DevOps-Paketen. Nach erfolgreichem Abschluss des Kurses sind die Studierenden dazu in der Lage

- reale Programmierprobleme zur objektorientierten Programmierung zu abstrahieren
- die Unterschiede und individuellen Vorteile der Programmierparadigmen der Objektorientierung, der funktionalen Programmierung und der logischen Programmierung verstehen
- in JAVA zu programmieren
- einfache Programme in Haskell und Prolog zu schreiben
- Moderne IDEs und DevOps-Pakete zu nutzen

Inhalt

- Grundlagen der Programmierung in JAVA
- Objektorientierte Programmierung in JAVA
- GUI-Programmierung in JAVA
- Entwurfsmuster in JAVA
- Grundlagen der funktionalen Programmierung in Haskell
- Grundlagen der Logikprogrammierung in Prolog

Lehrformen

Vorlesungen, digitale Übungen, und Übungsstunden

Prüfungsformen

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

6/158: B.Sc. Informatik

Titel des Moduls: Logik					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 2	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Logik: Vorlesung und Übung (211061)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Prof. Dr. Thomas Zeume					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik					
Vorkenntnisse Mathematik Grundlagenvorlesungen					
Lernziele (learning outcomes) In dieser Veranstaltung werden die formalen Grundlagen von modernen Logiken behandelt, mit einem Fokus auf ihrer Anwendung in der Informatik. Neben der klassischen Aussagenlogik und Prädikatenlogik betrachten wir auch Modallogik. Für jede dieser Logiken formalisieren wir Syntax und Semantik, lernen wie sich informatische Szenarien in ihnen modellieren lassen, und betrachten Algorithmen und Kalküle für Unerfüllbarkeit und Folgerungsbeziehung.					
Inhalt Logische Methoden spielen in vielen modernen Anwendungen der Informatik eine wichtige Rolle. Aus Datenbanken werden relevante Informationen mit Hilfe auf Logik basierender Anfragesprachen extrahiert; die formale Verifikation von Software und Hardware basiert auf logischen Spezifikationssprachen und Algorithmen für diese; und Methoden für das automatisierte Schlussfolgern in der künstlichen Intelligenz haben ihre Grundlage in der formalen Logik.					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur (90 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Statistik und Wahrscheinlichkeitsrechnung					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 3	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Höhere Mathematik C: Vorlesung (150104) Höhere Mathematik C: Übung (150105)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 400 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Herold Dehling Lehrende: Prof. Dr. Herold Dehling					
Verwendung des Moduls B.Sc. Angewandte Informatik [PO 22]					
Vorkenntnisse Inhalte aus den Modulen Höhere Mathematik 1 und 2					
Lernziele (learning outcomes) Die Studierenden <ul style="list-style-type: none"> • kennen die Grundbegriffe der Wahrscheinlichkeitstheorie und Statistik, • sind in der Lage, Standardaufgaben nachzuvollziehen und selbstständig zu bearbeiten, • kennen das Auftreten und die Bedeutung des Zufalls in Natur und Technik und • sind im Stande, Zufallsphänome mit Standardverfahren zu modellieren, • können das Erlernte auf konkrete ingenieurwissenschaftliche Problemstellungen anwenden 					
Inhalt Die Lehrveranstaltung behandelt das zum Verständnis und zur Modellierung von Zufallsphänomenen in den Ingenieurwissenschaften erforderliche Basiswissen der Wahrscheinlichkeitstheorie und der Statistik. Hierzu gehören im Bereich der Wahrscheinlichkeitstheorie: Modellierung von Zufallsexperimenten, Wahrscheinlichkeitsraum, bedingte Wahrscheinlichkeiten, Unabhängigkeit, diskrete und stetige Zufallsvariablen, Dichte- und Verteilungsfunktion, wichtige Wahrscheinlichkeitsverteilungen (u.a. binomial, Poisson, geometrisch, normal, exponentiell, Chi-Quadrat, F-Verteilung), Erwartungswert, Varianz, Kovarianz, Korrelationskoeffizient, gemeinsame Verteilung, Faltungsformel, sowie im Bereich der Statistik: Verfahren der beschreibenden Statistik, statistische Modellierung, Grundlagen der Schätztheorie (u.a. Maximum Likelihood Methode), Konfidenzintervalle, Grundlagen der Testtheorie, Fehler 1. und 2. Art, Niveau eines Tests, Tests bei normalverteilten Stichproben (t-Test, F-Test), Lineare Regressionsmodelle (Kleinste Quadrate Methode, t-Test), Chi-Quadrat-Test bei diskreten Daten, 1-Faktor ANOVA. Die Konzepte und Verfahren werden stets durch Anwendungsbeispiele und Simulationen mit Hilfe des statistischen Pakets R illustriert.					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur (90 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Informatik 3 Computer Science 3					
Modul-Nr./Code	Credits 8 CP	Workload 240 h	Semester 3	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Informatik 3 - Theoretische Informatik (212002)			Kontaktzeit 90 h	Selbststudium 150 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Prof. Dr. Thomas Zeume Prof. Julian Loss					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/ Informationstechnik					
Vorkenntnisse					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • beherrschen die Studierenden den professionellen Umgang mit Berechnungsmodellen und ihren Beziehungen zu Sprachklassen. Dazu gehört die intellektuelle und methodische Fähigkeit, den Nachweis der Zugehörigkeit bzw. Nichtzugehörigkeit zu einer solchen Klasse zu führen. • ist durch Einüben von Beweistechniken wie wechselseitige Simulation oder berechenbare Reduktionen bei den Studierenden die Einsicht gereift, dass an der Oberfläche verschieden aussehende Konzepte im Kern identisch sein können. Zudem erlaubt dies den Studierenden, neue Anwendungsprobleme selbstständig zu klassifizieren. • haben die Studierenden mit der Turingmaschine ein einfach handhabbares Rechnermodell erlernt, das ihnen fortan als Abstraktion für alle möglichen Rechner dient. • haben die Studierenden fundamentale Einsichten erlangt, welche Probleme mithilfe von Rechnern effizient entschieden, zum Teil entschieden oder prinzipiell nicht entschieden werden können. Dadurch erlangen Sie ein tieferes Verständnis der Komplexität von Berechnungsproblemen. 					
Inhalt Die Lehrveranstaltung gibt einen systematischen Überblick über die folgenden Themengebiete: <ul style="list-style-type: none"> • Endliche Automaten und reguläre Ausdrücke • Kellerautomaten und kontextfreie Grammatiken • Turingmaschinen und Entscheidbarkeit • Nichtdeterminismus und NP-Vollständigkeitstheorie 					
Lehrformen Hörsaalvorlesung mit Medienunterstützung und Übungen, bei denen die vorgestellten Konzepte und Techniken praktisch umgesetzt werden, teilweise mit Rechnerübungen.					
Prüfungsformen Klausur (180 Minuten)					

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

8/165: B.Sc. Informatik [PO 22]

8/158: B.Sc. Informatik [PO 22]

8/168: B.Sc. Angewandte Informatik [PO 22]

8/170: B.Sc. Angewandte Informatik [PO 20]

8/150: B.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

8/149: B.Sc. IT-Sicherheit/ Informationstechnik [PO 20]

Titel des Moduls: Software Engineering Software Engineering					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 3	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen 212000: Software Engineering			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 350 Studierende
Unterrichtssprache English			Teilnahmevoraussetzungen Programming and Programming Languages, Informatik 1		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thorsten Berger Lehrende: Prof. Dr. Thorsten Berger					
Verwendung des Moduls					
Vorkenntnisse Imperative and Object-Oriented Programming in a statically typed programming language Theoretical Computer Science					
Lernziele (learning outcomes)					
Knowledge and understanding					
<ul style="list-style-type: none"> • Explain in detail the core activities for engineering software, including requirements engineering, software architectures and design, implementation, quality assurance (esp. testing), software process • explain underlying representation of programs (AST) and the compilation process • explain programming paradigms, such as imperative programming, generic programming, object-oriented programming, AI engineering 					
Competence and skills					
<ul style="list-style-type: none"> • elicit and define software requirements in different formalisms (e.g., textual requirements, state machine diagrams, or other behavior diagrams) • define a software architecture upon quality requirements, using architectural patterns or styles • implement software architectures, frameworks, and software modules/components • define a software engineering process based on process models, including plan-based and agile models • perform quality assurance, including designing test cases according to test-case design methods for black-box and white-box testing • create behavioral and structural models in the context of software engineering 					
Judgement and approach					
<ul style="list-style-type: none"> • identify use-cases and the potential of different SE methods and technologies for a given domain/problem • select and justify SE methods and technologies for a given domain/problem 					
Inhalt					
<p>You can write code, but can you also write software? The course provides an introduction into Software Engineering methods and tools. Covering the overall phases of software engineering, namely planning, requirements engineering, architectural design, implementation, quality assurance, and evolution and maintenance, the curriculum will walk students through modern software engineering technology that aims at building the modern software systems and products. The course attempts to be close to programming technology, while covering multiple paradigms and solutions.</p> <p>We cover:</p> <ul style="list-style-type: none"> • Introduction to software engineering, motivated by software and project failures, laws of software engineering, and optionally a guest lecture from industry 					

- Introduction into compiler construction and the underlying representation of programs (e.g., abstract syntax trees, control-flow diagrams)
- Paradigms (e.g., imperative programming, generic programming, object-oriented programming, AI engineering) and software engineering principles (e.g., modularity, cohesion/coupling)
- UML behavioral and structural diagrams for modeling software
- Requirements engineering
- Software architecture and architecture implementation
- Quality assurance, including black-box and white-box testing, mutation testing, combinatorial interaction testing
- Advanced topics, such as software product lines, model-driven software engineering or security engineering (e.g., threat modeling)

Lehrformen

The teaching of this course consists of different forms: lectures, interactive quizzes, group work, group supervision, and practical assignments.

Prüfungsformen

Written exam at the end of the course (120 minutes)

Voraussetzungen für die Vergabe von Credits

Group project and final exam passed

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO20]

Titel des Moduls: Projektmanagement Project Management					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 3	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Projektmanagement (135040)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 150 Studierende
Unterrichtssprache Deutsch, Lehrmaterial teils auf Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Christian Meske Lehrende: Prof. Dr. Christian Meske					
Verwendung des Moduls B.Sc. Angewandte Informatik [PO 22]					
Vorkenntnisse					
Lernziele (learning outcomes) Studierende <ul style="list-style-type: none"> • verstehen die Rolle von Projektmanagement als Steuerungsinstrument im Spannungsfeld zwischen Zeit, Kosten und Qualität • verfügen über Grundkenntnisse und Fähigkeiten bezogen auf klassisches sowie agiles Projektmanagement • kennen alle relevanten Projektabläufe und -phasen sowie deren interdependenten Zusammenhänge • können theoretische Lehrinhalte (im Rahmen von Übungen) direkt anwenden • haben ihre Fähigkeit zur Teamarbeit verbessert 					
Inhalt Die Studierenden beschäftigen sich mit Methoden und Werkzeugen des Projektmanagements in klassischen und agilen Projektszenarien. Zu den Inhalten zählen u.a.: <ul style="list-style-type: none"> • Phasen eines Projekts • Projektmanagementprozess und Prozesselemente • Projekte aufsetzen und strukturieren • Rollen im Projekt, Projektführung • Elemente der Projektplanung und der Terminplanung • Qualitätsbegriff, Qualitätsmanagement in Projekten • Stakeholdermanagement • Risikomanagement • Projektumsetzung, Projektsteuerung und -controlling • Entscheidungen in Projekten • Projektabschluss 					
Lehrformen Die Veranstaltung findet über das Semester hinweg überwiegend in Präsenz statt und besteht sowohl aus einem Vorlesungs- als auch Übungsteil. Es werden klassische Vorlesungseinheiten stattfinden, ergänzt um asynchrone Online-Videos und weitere digitale Lehrinhalte. Parallel zur Vorlesung wird eine Übung angeboten (teils in Präsenz, teils virtuell) in der Studierende beispielhafte Aufgaben erhalten und bearbeiten. Zudem sind je nach Verfügbarkeit einzelne Gastvorträge aus der Praxis eingeplant.					
Prüfungsformen Klausur (60 Minuten)					

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

Titel des Moduls: Einführung in die künstliche Intelligenz Introduction to Artificial Intelligence					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Introduction to Artificial Intelligence (211045)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 250 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Sen Cheng Lehrende: Prof. Dr. Laurenz Wiskott, Prof. Dr. Tobias Glasmachers, Prof. Dr. Sen Cheng, Prof. Dr. Gregor Schöner, Prof. Dr. Nils Jansen Prof. Dr. Setareh Maghsudi Prof. Dr. Christian Straßer					
Verwendung des Moduls B.Sc. Informatik (Pflichtmodul) B.Sc. Angewandte Informatik (Pflichtmodul) B.Sc. IT-Sicherheit/Informationstechnik (Wahlpflichtmodul)					
Vorkenntnisse Basic knowledge of calculus and linear algebra.					
Lernziele (learning outcomes) After successful completion of this course, students will be able to <ul style="list-style-type: none"> • summarize a number of fundamental methods in artificial intelligence, • explain their mathematical basis and algorithmic nature, • apply them to simple problems, • decide which methods are suitable for which problems, and • communicate about the all that in English. 					
Inhalt This course gives an overview over representative methods in artificial intelligence: formal logic and reasoning, classical methods of AI, probabilistic reasoning, machine learning, deep neural networks, computational neuroscience, neural dynamics, perception, natural language processing, robotics.					
Lehrformen					
Prüfungsformen Written module final exam (digital, 90 minutes)					
Voraussetzungen für die Vergabe von Credits passed written exam					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/158: B.Sc. Informatik [PO 22] 5/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Datenbanksysteme Database Systems					
Modul-Nr./Code	Credits 7 CP	Workload 210 h	Semester 4	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Datenbanksysteme (211008)			Kontaktzeit 60 h	Selbststudium 150 h	Gruppengröße 250 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Veelasha Moonsamy Lehrende: Dr. Veelasha Moonsamy					
Verwendung des Moduls B.Sc. Informatik [PO 22] B.Sc. Angewandte Informatik [PO 22]					
Vorkenntnisse Grundkenntnisse der Informatik (Inhalte der Module Informatik 1; Programmierung, Technische Informatik 1; Rechnerarchitektur)					
Lernziele (learning outcomes) Nach dem erfolgreichen Absolvieren des Moduls <ul style="list-style-type: none"> • erlangen die Studierenden ein Grundverständnis von modernen Datenbanksystemen, ihrer Funktion und ihrer Implementierung • haben die Studierenden Datenmodellierungstechniken erlernt • haben die Studierenden die Semantik und die Syntax des Entity-Relationships Modells kennengelernt • kennen die Studenten das relational Datenbankmodell und die Relationale Algebra • kennen die Studierenden Anfragesprachen (z.B. SQL) und können diese nutzen • verstehen die Studierenden die Konzepte von Transaktion und Fehlerbehandlung • haben die Studierenden unterschiedliche Datenbankmanagementsysteme kennengelernt • sind die Studierenden in der Lage, neue Datenbanken zu modellieren und zu implementieren • haben die Studenten Kenntnisse über die Prozesse hinter einer Datenbankabfrage und wie diese optimiert werden kann 					
Inhalt Die Datenbanktechnologie ist eine Schlüsseltechnologie der praktischen und angewandten Informatik. Zentrales Thema dieser Veranstaltung sind die Modellierung, Aufbau und die Nutzung von Datenbanken. Folgende Themen werden behandelt: <ul style="list-style-type: none"> • Einführung in Datenbanksysteme • Entity-Relationship Modell und Verbesserungen • das relational Datenbankmodell • Relationale Algebra und Kalkül • Die Relationale Anfragesprache SQL • Datenbankprogrammierung • physische Datenorganisation • Anfragebearbeitung und Optimierung • Transaktionsverwaltung und Fehlerbehandlung 					
Lehrformen In der wöchentlichen Vorlesung werden die Lerninhalte theoretisch vermittelt. In der unterstützenden wöchentlichen Übung werden theoretische Fragestellungen sowie praktische Fragestellungen und Aufgaben am Computer bearbeitet. Die Aufgaben und Lösungen werden in der Übung gemeinsam erarbeitet und besprochen.					

Prüfungsformen

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

7/158: B.Sc. Informatik [PO 22]

7/168: B.Sc. Angewandte Informatik [PO 22]

Titel des Moduls: Web-Engineering Web-Engineering					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester Siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Web-Engineering (128968 + 128969)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 200 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Markus König Lehrende: Prof. Dr.-Ing. Markus König Stephan Embers, M.Sc.					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik M.Sc. IT-Sicherheit/ Informationstechnik					
Vorkenntnisse Programmieren, Architektur von Web-basierten Systemen					
Lernziele (learning outcomes) Die Entwicklung von Web-Anwendungen und Web-Services ist zentraler Bestandteil der Digitalisierung. Ziel der Lehrveranstaltung ist die Vermittlung von Grundlagen und bewährten Verfahren in der Web-Entwicklung. Studierende lernen konzeptuelle technologische Bausteine kennen: Transportverfahren, Webseitendarstellung, dynamische Web-Anwendungen und Web-Services. Über das konzeptuelle Verständnis hinaus werden praktische Kompetenzen vermittelt. Dazu werden moderne Werkzeuge der Web-Entwicklung, sowohl server- als auch clientseitig, vorgestellt und in den Übungssitzungen praktisch vertieft. Während der Umsetzung einfacher Web-Anwendungen stehen auch analytische Fähigkeiten im Fokus: Studierende werden befähigt, verschiedene Verfahren in Hinblick auf Performanz und Wartbarkeit zu bewerten. Diese Fähigkeiten sind in der kritischen Planungsphase von Software-Projekten unerlässlich. Nach dem erfolgreichen Abschluss des Moduls: <ul style="list-style-type: none"> • kennen Studierende gängige Konzepte der Web-Entwicklung in den Aspekten Präsentation, Transport und Bereitstellung von Daten • beherrschen Studierende grundlegende Fähigkeiten in Webseitendarstellung, dynamischen Web-Anwendungen und modernen Services (Node.js) 					
Inhalt Im Rahmen des Modules werden den Studierenden aktuelle Techniken und Kenntnisse im Bereich der Web-Entwicklung aufgezeigt. Thematisch wird der Bereich der server- und clientseitigen Entwicklung abgedeckt. JavaScript stellt dabei eine zentrale Rolle dar. Folgende Lehrinhalte werden behandelt: <ul style="list-style-type: none"> • Einführung in clientseitige Web-Entwicklung: HTML, CSS, JavaScript, Web Components • Transportverfahren und deren Nutzung: Representational State Transfer (REST), Asynchronous JavaScript und XML (AJAX) • Serverseitige Entwicklung mit Node.js und weiterführende Technologien 					
Lehrformen Synchrone Onlinevorlesung, Tutorien als seminaristischer Unterricht, zusätzlich Selbststudium mit ergänzend bereitgestellten Materialien und Aufgaben.					

Prüfungsformen

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/91: M.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

5/84: M.Sc. IT-Sicherheit/ Informationstechnik [PO 20]

Titel des Moduls: Betriebssysteme Operating Systems					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester 4	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Betriebssysteme (211005)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 350 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Timo Hönig Lehrende: Prof. Dr.-Ing. Timo Hönig					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/Informationstechnik					
Vorkenntnisse Grundkenntnisse der Informatik (Inhalte der Module Informatik 1; Programmierung und Technische Informatik 1; Rechnerarchitektur)					
Lernziele (learning outcomes) Nach dem erfolgreichen Absolvieren des Moduls <ul style="list-style-type: none"> • erlangen die Studierenden ein solides Grundverständnis von modernen Betriebssystemen, ihrer Funktion und ihrer Implementierung • sind die Studierenden in der Lage, verschiedene Aspekte eines Betriebssystems wie Prozess- und Speicheranagement zu verstehen und zu nutzen, sie können dabei verschiedene Designentscheidungen eigenständig analysieren und bewerten • sind die Studierenden in der Lage, bestimmte Aspekte eines Betriebssystems selbst zu designen und diese argumentativ zu verteidigen 					
Inhalt In diesem Modul werden die wichtigsten Grundlagen zu Betriebssystemen vorgestellt. Dazu gehören zum Beispiel: <ul style="list-style-type: none"> • Betriebssystemkonzepte • Prozesse und Threads, Interprozesskommunikation • Scheduling-Mechanismen • Speicherverwaltung, Speicherabstraktionen, Paging • Dateisysteme • Eingabe- und Ausgabeverwaltung • Algorithmen zur Vermeidung von Deadlocks • Grundlagen der Sicherheit von Betriebssystemen <p>In den letzten Wochen der Veranstaltung, abhängig vom verfügbaren Zeitfenster, werden spezielle Themen wie beispielsweise Multimedia-Betriebssysteme, Multiprozessorsysteme und Entwurf von Betriebssystemen, behandelt.</p> <p>Um den Bezug zu modernen Betriebssystemen (aktuellen Versionen von Linux, Windows und macOS) herzustellen, werden die Themen an praktischen Beispielen illustriert. Dies ermöglicht es den Studierenden, die in der Vorlesung besprochenen Themen praktisch nachzuvollziehen.</p>					
Lehrformen					

Die Vorlesung wird als seminaristischer Unterricht mit Medienunterstützung abgehalten. eLearning unterstützte Hausaufgaben mit praxisnahen, am Rechner zu implementierenden Übungen werden alle zwei Wochen vergeben und in der Übungsstunde besprochen.

Prüfungsformen

Klausur (90 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/170: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 20]

Titel des Moduls: Software Engineering Praktikum Software Engineering Lab					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Software Engineering Lab (211500)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 5 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thorsten Berger Lehrende: Prof. Dr. Thorsten Berger					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik					
Vorkenntnisse Programmiererfahrung in einer objektorientierten Programmiersprache (am besten Java), sicherer Umgang mit git					
Lernziele (learning outcomes) Nach dem Modulabschluss <ul style="list-style-type: none"> • können die Studierenden agil arbeiten nach SCRUM • wissen Studierende, wie man kleine Software-Projekte plant und können diese in Java-Android umsetzen • können Studierende ihre eigenen Ergebnisse in angemessener Form präsentieren 					
Inhalt Im Software Engineering Lab wird in kleinen Projektgruppen eine Android App mit AndroidStudio entwickelt. In der begleitenden Vorlesung werden die Grundlagen moderner Softwareentwicklung vermittelt und im Projekt praktisch umgesetzt. Die Projektgruppen arbeiten selbstorganisiert agil und werden durch den gesamten Entwicklungsprozess unterstützend angeleitet. Die Entwicklung beginnt mit der Backlog-Erstellung und endet mit einem kurzen Produkt-Pitch.					
Lehrformen Agiles Arbeiten in Projektgruppen					
Prüfungsformen Projektarbeit (semesterbegleitend) mit Zwischenmeetings, Abgaben und Abschlusspräsentation					
Voraussetzungen für die Vergabe von Credits Erfolgreich abgeschlossene Projektarbeit mit Abschlusspräsentation. Zum Bestehen des Moduls ist die Teilnahme am SCRUM-Workshop, an allen Supervisions- und Kundenterminen sowie an der Abschlusspräsentation (Prüfung) formal verpflichtend. Außerdem müssen Sie, wie in der ersten Vorlesung erläutert, eine Gruppe bilden und diese rechtzeitig bei uns anmelden. Die Anwesenheit bei den Vorlesungen ist dringend erwünscht, aber keine Pflicht.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/158: B.Sc. Informatik [PO 22] 5/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Datenschutz					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Datenschutz (260081)			Kontaktzeit 45 h	Selbststudium 105 h	Gruppengröße 120 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Kai-Uwe Loser Lehrende: Dr. Kai-Uwe Loser					
Verwendung des Moduls B. Sc. Angewandte Informatik B.Sc. IT-Sicherheit/ Informationstechnik					
Vorkenntnisse					
Lernziele (learning outcomes) Datenschutz befasst sich mit der Frage, wie man Bürger, Arbeitnehmer, Kunden, Patienten etc. vor negativen Auswirkungen durch die Verarbeitung von Daten zu ihrer Person schützen kann. Es besteht die Anforderung an Informatiker, Computersysteme so zu gestalten, dass sie die Umsetzung datenschutzrechtlicher Prinzipien unterstützen. Die Vorlesung befasst sich daher mit den Prinzipien des Datenschutzrechtes und den praktischen Auswirkungen für Informatiker. Dabei wird vor allem Wert darauf gelegt, diese zentralen Prinzipien verstehbar zu machen. Neben dem Datenschutzgrundverordnung werden auch Spezialregelungen behandelt, die z.B. für die Regulierung der Telekommunikation, oder für den Einsatz elektronischer Datenverarbeitung in der Arbeitswelt zum Einsatz kommen. Die DSGVO ist inzwischen auch über den europäischen Raum hinaus ein akzeptierter Standard. Unterschiedliche Rechtsphilosophische Betrachtungen werden thematisiert, um zu vermitteln, wo international Sichtweisen und Fragestellungen divergieren. Insgesamt wird das Thema konstruktiv betrachtet: das Thema Privacy by Design, wird auf allen Ebenen betrachtet. Lernziel der Vorlesung ist es, dass die Studierenden künftig in der Lage sind, zu erkennen, an welchen Stellen ihres beruflichen Wirkens der Datenschutz relevant ist, und wie sie vorgehen müssen, um sich geeignete Informationen oder Sachverstand zu besorgen. Das zu vermittelnde Wissen soll so grundlegend sein, dass man sich auch auf neue Entwicklungen (wie etwa Novellierungen und Ergänzungen des Bundesdatenschutzgesetzes) einstellen kann. Nach dem erfolgreichen Abschluss des Moduls					
<ul style="list-style-type: none"> • kennen Studierende die Grundzüge des Datenschutzrechtes, • verstehen Studierende die gesellschaftlichen Hintergründe, • können Datenverarbeitungsprozesse hinsichtlich der Relevanz des Datenschutzrechts analysieren und • können Lösungsmuster anwenden, um Systeme datenschutzfreundlich und datenschutzrechtskonform zu gestalten. 					
Inhalt					
<ul style="list-style-type: none"> • Was ist Datenschutz, informationelle Selbstbestimmung und Privacy? • Welche Folgen haben Verarbeitungen personenbezogener Daten? Woher entstehen diese Folgen? • Was sind die Prinzipien des Datenschutzes 					

- Welche Rechte haben die von der Verarbeitung betroffenen Personen?
- Was passiert mit personenbezogenen Daten in vernetzten Systemen?
- Welche organisatorischen und technischen Maßnahmen helfen, personenbezogene Daten zu sichern?
- Was ist Privacy by Design und wie kann das umgesetzt werden?
- Spezielle Bereiche der Datenverarbeitung: Telekommunikation, Wirtschaft, Medizin

Lehrformen

Vorlesung mit Folien, Übungen zu Wissens- und Verständnisabfragen sowie Anwendung auf Beispiele

Prüfungsformen

Klausur (90 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B. Sc. Angewandte Informatik [PO 22]

5/170: B. Sc. Angewandte Informatik [PO 20]

5/150: B.Sc. IT-Sicherheit/ Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/ Informationstechnik [PO 20]

Titel des Moduls: Algorithm Paradigms Algorithm Paradigms					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Algorithm Paradigms (211043)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 40 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Maïke Buchin Lehrende: Prof. Dr. Maïke Buchin					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik					
Vorkenntnisse Vorlesung über Algorithmen und Datenstrukturen					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende eine Reihe von Algorithmenparadigmen • können Studierende basierend auf den Paradigmen effiziente Algorithmen für Probleme entwickeln • verstehen Studierende die Vor- und Nachteile unterschiedlicher Paradigmen 					
Inhalt Die Vorlesung vertieft und ergänzt die Kenntnisse aus der Vorlesung "Informatik 2 - Algorithmen und Datenstrukturen". Konkret betrachten wir unterschiedliche Algorithmenparadigmen, also Schemata zum Entwurf von effizienten Algorithmen. Dazu betrachten wir zunächst die bereits bekannten Paradigma inkrementell, Teile-und-Herrsche und gierig und wenden diese auf verschiedene Probleme an. Darauf aufbauend lernen wir Dynamisches Programmieren kennen, sowie die Methoden Backtracking und Branch-and-Bound. Auch betrachten wir ein Paradigma speziell für geometrische Probleme: das Sweepline-Verfahren.					
Lehrformen Hörsaalvorlesung mit Medienunterstützung sowie Tutorien als seminaristischer Unterricht					
Prüfungsformen Schriftliche Klausur (120 Minuten) oder mündliche Prüfung (15-45 min)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur (120 min.) oder mündliche Modulabschlussprüfung (15-45 min)					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/170: B.Sc. Informatik [PO 22] 5/158: B.Sc. Informatik [PO 20] 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Bioinformatics for Molecular and Cellular Structures					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Lecture: Bioinformatics for Molecular and Cellular Structures (190702) Exercise for Lecture: Bioinformatics for Molecular and Cellular Structures (190703)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Axel Mosig Lehrende: Prof. Dr. Axel Mosig, Prof. Dr. Raphael Stoll, Dr. Daniel Todt					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Grundkenntnisse in (Molekularer) Biologie.					
Lernziele (learning outcomes) Die Teilnehmer/innen erwerben Grundkenntnisse und erhalten einen Einblick in die aktuellen Werkzeuge und zugrunde liegenden Methoden der Bioinformatik. Erworbene Kompetenzen liegen vor allem im Bereich des Erlernens bioinformatischer Werkzeuge, des Identifizierens angemessener Bioinformatik Methoden für biologische Fragestellungen sowie das Erlernen von formalem mathematisch-informatischen Denkens. Hierbei spielt das Erlernen interdisziplinären Denkens und das Anwenden von Fähigkeiten und Wissen über Fächergrenzen hinweg eine besondere Rolle.					
Inhalt Bioinformatische Werkzeuge und Methoden sind zu einem festen Bestandteil der biologischen Forschung geworden. Die Vorlesung gibt einen Überblick über die aktuellen Werkzeuge und die zugrunde liegenden Methoden, mit einem besonderen Schwerpunkt der Analyse von Sequenz und Struktur von Proteinen. In den Übungen wird einerseits die Anwendung dieser Werkzeuge in der Praxis vermittelt, andererseits die theoretischen Grundlagen anhand von Übungsaufgaben vertieft.					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur (120 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Computerlinguistische Programmierung					
Modul-Nr./Code	Credits 9 CP	Workload 270h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen			Kontaktzeit 60h (4 SWS)	Selbststudium 210h	Gruppengröße 10 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Das Modul kann nur belegt werden, wenn parallel „Einführung in die Computerlinguistik“ belegt wird.		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Stefanie Dipper Lehrende: Prof. Dr. Stefanie Dipper, Dr. Ronja Laarmann-Quante					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Grundlegendes Wissen über die wichtigsten Konzepte, Modelle und Methoden aus der Statistik und Programmierung.					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls sind die Studierenden in der Lage, komplexe Modelle und Algorithmen zur Analyse von Sprachdaten selbst zu implementieren. <ul style="list-style-type: none"> • Sie können zu einem vorgegebenen Pseudocode ein entsprechendes Programm implementieren, das den Pseudocode umsetzt. • Sie können Beschreibungen von Verfahren, wie sie in Lehrbüchern oder computerlinguistischen Fachartikeln präsentiert werden, in Form von Pseudocode wiedergeben. • Sie können Methoden und Algorithmen auf eigene Fragestellungen anwenden. 					
Inhalt Dieses Modul umfasst eine praktische Übung zum Aufbaumodul Computerlinguistik, in dem ausgewählte theoretische Inhalte aus dem Aufbaumodul praktisch umgesetzt werden. Die Studierenden implementieren grundlegende Algorithmen und typische Aufgabenstellungen der Computerlinguistik nach. Sie lernen Best Practices beim Umsetzen komplexer Programme kennen und erhalten individuelle Unterstützung beim Programmieren. Vor- und Nachteile ausgewählter (Teil-)Lösungen werden exemplarisch besprochen.					
Lehrformen Praktische Übung.					
Prüfungsformen Implementation mit Dokumentation.					
Voraussetzungen für die Vergabe von Credits Erfolgreicher Abschluss der Modulprüfung sowie bestandene Studienleistung: Portfolio aus Implementationsaufgaben, Präsentationen. In der Lehrveranstaltung steht die Diskurs-, Handlungs- oder Praxisorientierung im Vordergrund. Daher besteht im Allgemeinen Anwesenheitspflicht.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 9/168: B.Sc. Angewandte Informatik					

Titel des Moduls: Digitaltechnik					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Technische Informatik 2 - Digitaltechnik für ITS und Informatik (211014)			Kontaktzeit 60 h	Selbststudium 90	Gruppengröße Studierende
Unterrichtssprache Deutsch, Vorlesungs- und Übungsfolien auf Englisch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tim Güneysu Lehrende: Prof. Dr. Amir Moradi					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Inhalte des Moduls Mathematik 1; Grundlagen bzw. Höhere Mathematik 1 und 2. Vorausgesetzt wird ein generelles Interesse an technischen Systemen, die Fähigkeit zu strukturieren, algorithmischem Denken sowie die Fähigkeit zum Erfassen von komplexen Abhängigkeiten und Interaktionsmustern.					
Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls haben die Studierenden umfassende Kenntnisse in Boolescher Algebra, Struktur und Funktionsweise grundlegender digitaler Schaltungen, Kostenoptimierung digitaler Funktionsgruppen, Techniken zur taktsynchronen Verarbeitung von Daten, Kodierung und Verarbeitung von Daten, Struktur und Funktionsweise solcher Grundfunktionalitäten, die insbesondere in Mikroprozessorarchitekturen zentrale Bestandteile sind, erworben. Die Studierenden sind in der Lage, grundlegende Schaltungskonzepte digitaler Logik- und Funktionsblöcke zu verstehen, ihr Zusammenspiel zu analysieren, die Funktionalität zu bewerten und einfache Blöcke selbst zu entwickeln. Weiterhin werden die Bewertung und Entwicklung von mehrstufigen kombinatorischen Logikblöcken sowie von Finite State Machines (FSMs) behandelt. Die Studierenden erlernen die Hardwarebeschreibungssprache Verilog, und zu jedem Thema der Vorlesung werden Verilog-Beispiele gegeben. Die Vorlesung befasst sich ausschließlich mit (takt-)synchronen Schaltungen.					
Inhalt Der Kurs gibt einen systematischen Überblick über die folgenden Themen: Boolesche Algebra, Realisierung boolescher Funktionen, Minimierung boolescher Funktionen, Multiplexer, Kodierer, Dekodierer, fehlererkennende und fehlerkorrigierende Codes, Addierer, Subtrahierer, Multiplizierer, Hardwarebeschreibungssprache Verilog, Speicherelemente (Flipflops), sequentielle Schaltungen, Zähler, Schieberegister, RAM, Finite State Machines (FSMs), Timing-Analyse sequentieller Schaltungen, und kurzer Überblick über FPGAs.					
Lehrformen Die Vorlesung wird als seminaristischer Unterricht abgehalten, die Übungen entweder am Rechner oder mit Stift und Papier.					
Prüfungsformen Klausur (120 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur und erfolgreiche Teilnahme an Übungen					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Distributed Systems Algorithms
Distributed Systems Algorithms

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Distributed Systems Algorithms (211004)			Kontaktzeit 45 h (3 SWS)	Selbststudium 105h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		

Modulbeauftragte/r und hauptamtlich Lehrende
 Modulbeauftragte/r: Prof. Dr.-Ing. Steffen Bondorf
 Lehrende: Prof. Dr.-Ing. Steffen Bondorf

Verwendung des Moduls

B.Sc. Angewandte Informatik
 B.Sc. Informatik

Vorkenntnisse

Allgemeines Interesse an technischen Systemen

Lernziele (learning outcomes)

Die Studierenden sollen eine breite Kenntnis über die auftretenden Herausforderungen beim Entwurf und bei der Anwendung von verteilten Computersystemen erlangen. Nach dem erfolgreichen Abschluss des Moduls

- kennen die Studierenden verschiedene Systemmodelle und Architekturen, die zum Entwurf sowie zu der Klassifizierung von verteilten Systemen dienen. Sie können verschiedene Rollen von Teilsystemen differenzieren und sie formal beschreiben
- können die Studierenden vielfältige Herausforderungen beim Aufbau eines verteilten Systems identifizieren und kennen die wichtigsten Standardtechniken zum Umgang mit diesen, inklusive deren Vor- und Nachteile
- können die Studierenden die Funktionsweise eines verteilt implementierten Systems anhand dessen Beschreibung verstehen und die ausgeführte Aufgabe herleiten - können die Studierenden die Fähigkeit eines verteilten Systems zur Erfüllung seiner Aufgabe beurteilen, die Quellen potenzieller Probleme identifizieren und können Verbesserungen sowie deren Integration entwerfen
- sind die Studierenden in der Lage, gegebene Alternativen zur verteilten Implementierung eines Systems für eine bestimmte Aufgabe zu bewerten und begründet in eine Rangfolge zu bringen

Inhalt

Diese Lehrveranstaltung behandelt grundlegende Architekturen und Methoden, die die Funktionsfähigkeit leistungsfähiger verteilter Computersysteme ermöglichen. Ein solches verteilte System dient der Erfüllung einer bestimmten Aufgabe unter Verwendung von mehreren unabhängigen Teilsystemen und soll dem Benutzer dabei jedoch wie ein einzelnes Computersystem erscheinen. Um dies zu erreichen, müssen die verschiedenen Teilsysteme über gemeinsames Wissen verfügen. Es treten durch die Verteilung im Vergleich zu einzelnen Systemen eine Reihe von Herausforderungen auf, die den Inhalt der Vorlesung bilden: Teilsysteme müssen sich gegenseitig auffinden können, sie müssen in der Lage sein, Nachrichten auszutauschen, Daten müssen trotz der so entstehenden Replikation über Teilsysteme hinweg konsistent gehalten werden, Fehler in Teilsystemen müssen tolerierbar sein und die Ressourcen des Gesamtsystems sollen möglichst effizient genutzt werden, sodass die gegebene Aufgabe performant erfüllt wird. All diese Komponenten und Aspekte finden sich in modernen, Internetbasierten Systemen wieder. Sie garantieren die Funktionsfähigkeit von Diensten wie das World Wide Web, E-Mail oder File-Sharing.

Lehrformen

Die Vorlesung wird als seminaristischer Unterricht abgehalten, die praktischen Übungen am Rechner werden zudem weitere Lehrformen wie Gruppen- und Projektarbeit beinhalten

Prüfungsformen

Klausur (90 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur und erfolgreiche Teilnahme an Übungen

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik

5/158: B.Sc. Informatik

Titel des Moduls: Einführung in die Computerlinguistik					
Modul-Nr./Code	Credits 6 CP	Workload 180h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen			Kontaktzeit 60h (4 SWS)	Selbststudium 120h	Gruppengröße 20 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Ronja Laarmann-Quante Lehrende: Prof. Dr. Stefanie Dipper, Dr. Ronja Laarmann-Quante					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Grundlegendes Wissen über die wichtigsten basalen Konzepte und Methoden aus der Statistik und der Linguistik.					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls verfügen die Studierenden über grundlegendes Wissen über Computerlinguistik: <ul style="list-style-type: none"> • Sie kennen die wichtigsten, grundlegenden Analyseebenen in der Computerlinguistik und ihre spezifischen Eigenschaften. • Sie kennen verschiedene, insbesondere statistische Verfahren zur automatischen Analyse von Sprachdaten sowie die dazugehörigen effizienten Algorithmen. • Sie kennen die spezifischen Probleme bei der automatischen Verarbeitung sprachlicher Daten (wie z. B. sparse data, Ambiguitäten) und können diese anhand konkreter Beispiele erklären. • Sie kennen eine Reihe von Evaluationsverfahren sowie ihre Vor- und Nachteile und können diese anwenden. • Sie kennen relevante Ressourcen und wissen, in welcher Form sie eingesetzt werden. 					
Inhalt Dieses Modul führt in die Kernmethoden und -modelle der computerlinguistischen Forschung ein. Neben theoretischen Grundlagen der Computerlinguistik (wie der Chomsky-Hierarchie) werden entsprechende Modellierungen der verschiedenen Komplexitätsebenen eingeführt. Dabei werden vorrangig statistische Verfahren behandelt, bei denen das System Informationen aus Daten lernt. Neben klassischen probabilistischen Verfahren werden auch Neuronale Modelle behandelt, die in den letzten Jahren vermehrt Anwendung in der maschinellen Sprachverarbeitung finden.					
Lehrformen Seminar mit Anteilen von Inverted Classroom Settings mit digital verfügbaren, u.a. videobasierten Lehrmaterialien zum Selbststudium.					
Prüfungsformen Schriftliche Aufgabe mit Präsentation.					
Voraussetzungen für die Vergabe von Credits Erfolgreicher Abschluss der Modulprüfung sowie bestandene Studienleistung: Portfolio aus Lektüreaufgaben, schriftlichen Hausaufgaben, Diskussionsbeiträgen.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 6/168: B.Sc. Angewandte Informatik					

Titel des Moduls: Einführung in die Kryptographie 1
Introduction to Cryptography 1

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Einführung in die Kryptographie 1 (212010)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 300 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen keine		

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr.-Ing. Christof Paar
 Lehrende: Prof. Dr.-Ing. Christof Paar

Verwendung des Moduls

B.Sc. IT-Sicherheit
 B.Sc. Informatik
 B.Sc. Angewandte Informatik
 M.Sc. IT-Sicherheit/ Netze und Systeme

Vorkenntnisse

Fähigkeit zum abstrakten und logischen Denken.

Lernziele (learning outcomes)

Nach erfolgreichem Abschluss des Moduls verfügen die Studierenden über Kenntnisse der symmetrischen und asymmetrischen kryptographischer Verfahren. Sie können entscheiden, unter welchen Bedingungen man in der Praxis bestimmte Chiffren einsetzt und wie die Sicherheitsparameter zu wählen sind. Sie sind mit den mathematischen Grundlagen, auf denen aktuelle Kryptoverfahren beruhen, vertraut. Durch die Beschreibung ausgewählter praxisrelevanter Chiffren, wie z. B. des AES- oder RSA-Algorithmus sowie des Diffie-Hellman-Schlüsselaustauschs, erreichen die Studierenden zudem ein algorithmisches und technisches Verständnis für moderne Kryptographie. Sie sind in der Lage argumentativ eine bestimmte Lösung zu verteidigen. Die Vorlesungen werden zusätzlich auch als Videos in Deutsch und Englisch angeboten. Durch das zweisprachige E-Learning-Angebot können die Studierenden Sprachkompetenzen in der Wissenschaftssprache Englisch erwerben.

Inhalt

Das Modul bietet eine Einführung in die moderne angewandte Kryptographie und Grundlagen der IT-Sicherheit. Der Fokus liegt auf kryptographischen Verfahren, der hierfür notwendigen Mathematik sowie dem Zusammenspiel von Kryptographie und IT-Sicherheit. Es werden viele für die Anwendung relevante symmetrische und asymmetrische kryptographische Verfahren vorgestellt und an praxisrelevanten Beispielen erläutert.

Die Vorlesung lässt sich in zwei Teile gliedern: Die Funktionsweise der symmetrischen Kryptographie, einschließlich der Beschreibung historischer symmetrischer Chiffren (Caesar Chiffre, affine Chiffre, One Time Pad) und aktueller symmetrischer Verfahren (Advanced Encryption Standard, Data Encryption Standard, Stromchiffren), wird im ersten Teil behandelt.

Der zweite Teil besteht aus einer Einführung in die asymmetrische Kryptographie und zwei ihrer wichtigsten Stellvertreter (RSA und der Diffie-Hellman-Schlüsselaustausch). Hierfür werden relevante Grundlagen der Zahlentheorie eingeführt, um ein grundlegendes Verständnis der Verfahren sicherzustellen (u. a. Ringe ganzer Zahlen, Gruppen, Körper, diskrete Logarithmen, euklidischer Algorithmus, eulersche Phi-Funktion). Nichtsdestotrotz liegt der Schwerpunkt auf der algorithmischen Einführung asymmetrischer Krypto-Verfahren.

Lehrformen

Vorlesung mit Übung, die Veranstaltung wird digital angeboten

Prüfungsformen

Klausurarbeit (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Erfolgreiches Bestehen der Modulklausur.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/149: B.Sc. IT-Sicherheit [PO 20]

5/150: B.Sc. IT-Sicherheit [PO 22]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/96 : M.Sc. IT-Sicherheit/ Netze und Systeme [PO20]

5/99 : M.Sc. IT-Sicherheit/ Netze und Systeme [PO22]

Titel des Moduls: Einführung in die Kryptographie 2
In-tro-duc-tion to Cryp-to-gra-phy 2

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Einführung in die Kryptographie 2 (211009)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 300 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr.-Ing. Christof Paar
 Lehrende: Prof. Dr.-Ing. Christof Paar

Verwendung des Moduls

B.Sc. IT-Sicherheit
 B.Sc. Informatik
 B.Sc. Angewandte Informatik
 M.Sc. IT-Sicherheit/ Netze und Systeme

Vorkenntnisse

Inhalte der Vorlesung "Einführung in die Kryptographie 1"

Lernziele (learning outcomes)

Nach erfolgreichem Abschluss des Moduls verfügen die Studierenden über Kenntnisse der grundlegenden Anwendungen asymmetrischer und hybrider Verfahren. Sie können entscheiden, wann welche kryptographischen Lösungen eingesetzt werden können, um gewisse Sicherheitsziele zu erreichen. Die Studierenden erreichen durch Beschreibungen ausgewählter praxisrelevanter Algorithmen, z. B. basierend auf elliptischen Kurven, Hash-Funktionen, digitalen Signaturverfahren und Post-Quantum-Kryptographie, ein algorithmisches und technisches Verständnis für den Einsatz in der Praxis. Sie wissen, wie Sicherheitsparameter für kryptographische Lösungen zu wählen sind und sind in der Lage argumentativ eine bestimmte Lösung zu verteidigen. Die Vorlesung bereitet zudem auf Veranstaltungen vor, in denen Kryptographie eingesetzt wird, beispielsweise in der Netz- und Software-Sicherheit, sowie auf Vorlesungen, in denen Kryptographie mit formalen Methoden behandelt wird. Die Vorlesungen werden zusätzlich auch als Videos in Deutsch und Englisch angeboten. Durch das zweisprachige E-Learning-Angebot können die Studierenden Sprachkompetenzen in der Wissenschaftssprache Englisch erwerben.

Inhalt

Das Modul bietet eine vertiefende Einführung in die moderne angewandte Kryptographie und Grundlagen der IT-Sicherheit. Der Fokus liegt auf asymmetrischen kryptographischen Verfahren, Hashfunktionen und der Konstruktion von Schemata wie digitalen Signaturen und Protokollen zur Schlüsselvereinbarung.

Die Vorlesung lässt sich in drei Teile gliedern: Im ersten Teil werden Verfahren basierend auf elliptischen Kurven, ein für die Praxis sehr wichtiger Vertreter der asymmetrischen Kryptographie, vorgestellt. Ebenso werden die Grundlagen der sogenannten Post-Quanten-Kryptographie erläutert und Hash-basierte Verfahren eingeführt.

Im zweiten Teil der Vorlesung werden Hash-Funktionen, die streng genommen symmetrische Verfahren sind, behandelt. Als wichtiger Praxisvertreter wird der Algorithmus SHA-3 im Detail besprochen.

Im letzten Teil werden Sicherheitsdienste (u. a. Authentisierung, Integrität und Nichtzurückweisbarkeit) und hierfür notwendige Schemata behandelt, die auf symmetrischen und asymmetrischen Krypto-Verfahren beruhen. Konkret

werden Schemata für digitale Signaturen, Message Authentication Codes (MACs) sowie die Grundlagen der Schlüsselvereinbarung, digitale Zertifikate und PKI eingeführt.

Lehrformen

Vorlesung mit Übungen, Die Veranstaltung wird online durchgeführt

Prüfungsformen

Klausurarbeit (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 20]

5/165: B.Sc. Informatik [PO 22]

5/158: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/96 : M.Sc. IT-Sicherheit / Netze und Systeme [PO20]

5/99 : M.Sc. IT-Sicherheit / Netze und Systeme [PO22]

Titel des Moduls: Einführung in die Linguistik					
Modul-Nr./Code	Credits 8 CP	Workload 240 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Einführung in die Linguistik (050004)			Kontaktzeit 90 h	Selbststudium 150 h	Gruppengröße 20 Studierende
Unterrichtssprache Deutsch und Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Ronja Laarmann-Quante Lehrende: Prof. Dr. Eva Belke, Prof. Dr. Stefanie Dipper, Prof. Dr. Agata Renans					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls verfügen die Studierenden über grundlegendes Wissen und elementare analytische/methodische Fertigkeiten in den zentralen Teildisziplinen der Linguistik (Phonetik/Phonologie, Morphologie, Syntax, Semantik, Pragmatik). Sie können <ul style="list-style-type: none"> • die grundlegenden Begriffe und Konzepte der Teildisziplinen sowohl auf Deutsch als auch auf Englisch auflisten, dabei terminologisch korrekt benennen und darstellen. • sprachwissenschaftliche Sachverhalte und Prozesse zutreffend bezeichnen, aufzählen und definieren. • die elementaren analytischen Methoden der strukturellen Linguistik in Phonetik/Phonologie, Morphologie, Syntax, Semantik und Pragmatik identifizieren und die entsprechenden Prozeduren skizzieren. • elementare Darstellungen und Analysen klassifizieren und den verschiedenen linguistischen Teildisziplinen und Forschungsgebieten zuordnen. • erste, einfache Analysen sprachlicher Daten auf den Beschreibungsebenen der Phonetik/Phonologie, Morphologie, Syntax, Semantik und Pragmatik durchführen und können dafür Methoden je nach analytischer Aufgabenstellung richtig auswählen, zuordnen und anwenden. 					
Inhalt Das Modul vermittelt die grundlegenden Konzepte und Strukturmerkmale der zentralen linguistischen Teildisziplinen bzw. Beschreibungsebenen (Phonetik/Phonologie, Morphologie, Syntax, Semantik, Pragmatik).					
Lehrformen <ul style="list-style-type: none"> • Präsentation durch Kursleiter:in mit Fragen an die Teilnehmer:innen und Diskussion einzelner Aspekte. • Kleinere Übungen bzw. Analysen während der Sitzungen. • Praktische Übungssitzungen. 					
Prüfungsformen Die benotete Modulabschlussprüfung wird in Form von schriftlichen Hausaufgaben in diesem Grundkurs sowie einer abschließenden Klausur voraussichtlich gegen Ende der Vorlesungszeit erbracht. Die genauen Bedingungen dieser Prüfungsleistung bzw. dieser Modulabschlussprüfung werden frühzeitig im Grundkurs bekannt gegeben.					
Voraussetzungen für die Vergabe von Credits <ol style="list-style-type: none"> 1. Bestandene Modulabschlussklausur (gegen Vorlesungsende) 					

2. Bestandene Hausarbeiten (Studienleistung) während der Vorlesungszeit

Wichtig:

In der Lehrveranstaltung steht die Diskurs-, Handlungs- oder Praxisorientierung im Vordergrund. Daher besteht im Allgemeinen Anwesenheitspflicht.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

8/168: B.Sc. Angewandte Informatik [PO 22]

Titel des Moduls: Fertigungsautomatisierung					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Fertigungsautomatisierung (136460)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Bernd Kuhlenkötter Lehrende: Dr.-Ing. Jannis Sinnemann					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Die vorherige erfolgreiche Teilnahme am Modul Grundlagen der Automatisierungstechnik ist keine Voraussetzung, aber ggf. hilfreich.					
Lernziele (learning outcomes) Aufbauend auf den in den Grundlagen der Automatisierungstechnik vermittelten Kenntnissen lernen die Studierenden vertiefte ingenieurwissenschaftliche Grundlagen im Bereich der automatisierten Fertigungsverfahren mit bahngesteuerten Werkzeugen kennen. Dabei sollen vier zentrale Lernziele auf Modulebene erreicht werden: <ul style="list-style-type: none"> • Die Studierenden verstehen den technischen Aufbau, die Einsatzmöglichkeiten sowie Vor- und Nachteile von unterschiedlichen Automatisierungskomponenten. Dieses Wissen können die Studierenden anwenden, um für einen Fertigungsprozess geeignete Automatisierungskomponenten auszuwählen. • Für Roboter/Bewegungsautomaten können kinematische Ketten abgeleitet und die Koordinatensysteme der einzelnen Achsen dargestellt werden. Es können Koordinatentransformationen von der Basis bis zum Endeffektor abgeleitet und ein Steuerungsprogramm entworfen werden. • Bei Greiftechnik können die Studierenden die wirkenden Kräfte und Beschleunigungen ermitteln. Zudem können Greifer in Bezug auf deren Wirkprinzipien analysiert werden. • Für gegebene Fertigungsprobleme können die Studierenden eine Wirtschaftlichkeitsbetrachtung erstellen sowie Kosten und Risiken ableiten. Zudem können bekannte Lösungen auf andere Automatisierungsprobleme übertragen und geeignete Lösungskonzepte geplant werden. 					
Inhalt In der Veranstaltung werden zunächst unterschiedliche Fertigungsverfahren (z. B. Roboter in der spanenden Fertigung, Roboter in Umformprozessen) vorgestellt. Im Fokus stehen dabei die Automatisierungsmöglichkeiten der jeweiligen Verfahren. Anhand von praxisnahen Beispielen werden die spezifischen Anforderungen an die Automatisierung herausgearbeitet. Nachfolgend werden Roboter/ Bewegungsautomaten diskutiert. Dabei wird das gesamte Spektrum vom Aufbau unterschiedlicher Getriebe, über einzelne Gelenke bis zum Aufbau von Industrierobotern behandelt und es werden Vor- und Nachteile der einzelnen Automaten, bspw. in Bezug auf deren Genauigkeit, erarbeitet. Ein weiterer Abschnitt behandelt die für Werkzeugmaschinen und Roboter wichtigen Wegmesssysteme. Mit Blick auf eine vollständige Fertigungsautomatisierung wird im Anschluss auf Materialfluss- und Ordnungseinrichtungen sowie auf Endeffektoren eingegangen. Dabei liegt ein Fokus auf der Greiftechnik, welche eine Handhabung von Werkstücken ermöglicht und gemäß der Vielfalt an Werkstücken ein großes Spektrum an unterschiedlichen Greifprinzipien und entsprechenden Umsetzungen bietet. Im Weiteren werden unterschiedliche Fertigungskonzepte und -systeme diskutiert. Dabei werden sowohl Gründe und Möglichkeiten für die Automatisierung erläutert als auch Vor- und Nachteile unterschiedlicher Fertigungslösungen aus technischer und wirtschaftlicher Sicht abgeleitet. Die einzelnen Aspekte werden anhand realer Beispiele verdeutlicht.					
Lehrformen Vorlesung mit Übung					

Prüfungsformen

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Functional Programming
Functional Programming

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Functional Programming (211060)			Kontaktzeit 4 SWS (60 h)	Selbststudium 90 h	Gruppengröße 50 Studierende

Unterrichtssprache Englisch	Teilnahmevoraussetzungen kein
---------------------------------------	---

Modulbeauftragte/r und hauptamtlich Lehrende
 Modulbeauftragte/r: Dr. Clara Schneidewind
 Lehrende: Dr. Clara Schneidewind
 Dr. Jana Hofmann
 Dr. Catalin Hritcu

Verwendung des Moduls
 B.Sc. Informatik

 B.Sc. Angewandte Informatik

 M.Sc. IT-Sicherheit / Informationstechnik

Vorkenntnisse
 Es sind keine besonderen Vorkenntnisse erforderlich.

- Lernziele (learning outcomes)**
 After successful completion of this course, students will:
- develop programs in high-level, functional programming languages, in particular OCaml
 - understand and apply the use of recursion to define data structures (lists, maps, trees, etc.) and purely functional algorithms
 - understand the structure and advantages of type systems and use them to support program design and implementation
 - study advanced functional programming features, including type polymorphism and higher-order functions
 - reason informally about the correctness and efficiency of functional programs and be aware of more formal alternatives to reasoning
 - apply type abstraction and modularization to structure programs into collections of libraries and use those to build more complex programs on top of them
 - argue about the correctness and security of functional programs
 - understand the fundamental principles of programming language design, especially applied to functional programming
 - design and develop simple programming languages, covering their formal definition and subsequent implementation as interpreters

Inhalt
 This course offers a rigorous and hands-on introduction to the principles and practice of functional programming—an increasingly influential paradigm that underpins the development of reliable, maintainable, and secure software.

 Functional programming centers on the use of pure functions: self-contained computations that produce outputs without altering external state. This absence of side effects leads to code that is not only elegant and expressive but also inherently more predictable and easier to reason about than traditional imperative code. As a result,

functional programming is particularly well-suited for building security-critical systems, where correctness and robustness are paramount.

The course uses OCaml, a modern functional language, to explore the theory and practice of functional programming from the ground up. Students will learn how strong static typing and advanced type systems serve as powerful tools for writing safe, composable, and error-resistant programs.

Through a mix of lectures, interactive exercises, and extended case studies (including the design and implementation of a small programming language), students will gain a deep understanding of how functional languages work and why they matter. These skills translate seamlessly to many modern mainstream languages, which increasingly incorporate functional concepts.

In addition to mastering practical programming techniques, students will also learn to reason formally about program correctness—an essential competency for those aiming to build high-assurance systems. By the end of the course, students will be equipped not just to write functional programs, but also to argue about their security and correctness.

Lehrformen

Vorlesung mit Übung

Prüfungsformen

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestehen der Klausur und erfolgreiche Teilnahme an den Übungen.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/170: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/91: M.Sc. IT-Sicherheit / Informationstechnik [PO 22]

5/84: M.Sc. IT-Sicherheit / Informationstechnik [PO 20]

Titel des Moduls: Geometrische Modellierung und Visualisierung Geometric Modeling and Visualization					
Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Geometrische Modellierung und Visualisierung (129008)			Kontaktzeit 60 h	Selbststudium 120 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Markus König Lehrende: Prof. Dr.-Ing. Markus König					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes) Die Studierenden <ul style="list-style-type: none"> • erarbeiten wesentliche geometrische Methoden im Ingenieurwesen und wenden diese zielgerichtet auf Ingenieuraufgaben an, • sind in der Lage, verschiedene geometrische Aufgabenstellungen aus Forschung und Praxis unter Verwendung aktueller Methoden der Mathematik und Informatik zu lösen, • können die theoretischen Grundlagen der behandelten Ansätze anwenden und Forschungsergebnisse aus diesen Bereichen reflektieren und beurteilen. 					
Inhalt <ul style="list-style-type: none"> • Geometrische Modelle • Affine Abbildungen und Differentialgeometrie • Freiformkurven und Freiformflächen • Boundary Representation • Constructive Solid Geometry • Octrees • Zerlegung und Triangulierung • Visualisierungstechniken 					
Lehrformen Vorlesung mit Übung am PC					
Prüfungsformen Klausur (180 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 6/168: B.Sc. Angewandte Informatik [PO 22] 6/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Geschäftsprozess-Management					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Vorlesung: Geschäftsprozess-Management (074170) Übung: Geschäftsprozess-Management (074171)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 40 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Marion Steven Lehrende: Dr. Roland Düsing					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes) Die TeilnehmerInnen des Kurses sollen <ul style="list-style-type: none"> • die funktionsorientierte und prozessorientierte Organisation von Unternehmen unterscheiden und bewerten können, • die Zielsetzung, Phasen und Ausrichtungen des Geschäftsprozess-Managements erklären können, • die Grundsätze, Sprachen und Werkzeuge der Modellierung von Geschäftsprozessen charakterisieren können, • auf der Grundlage einer Modellierungssprache Geschäftsprozess- Modelle einfacher Komplexität entwickeln können, Geschäftsprozess-Modelle im Hinblick auf verschiedene Kriterien wie z. B. Prozesszeit, Prozessqualität oder Prozesskosten bewerten und verbessern können. 					
Inhalt Geschäftsprozess-Management ermöglicht eine zielgerichtete Steuerung der Geschäftsprozesse eines Unternehmens. Ein Geschäftsprozess, wie z. B. die Bearbeitung einer Kundenanfrage oder der Verkauf eines Produkts, besteht aus einer zusammenhängenden abgeschlossenen Folge von Tätigkeiten, die auf die Erfüllung der Bedürfnisse von Kunden ausgerichtet ist. Gleichzeitig sollen Geschäftsprozesse die strategischen und operativen Zielvorgaben erfüllen, von denen der Geschäftserfolg eines Unternehmens abhängt. In diesem Modul werden die Grundlagen des Geschäftsprozess-Managements erläutert. Dabei liegt der Schwerpunkt auf der Modellierung von Geschäftsprozessen. Es wird eine ausgewählte Sprache zur Geschäftsprozess-Modellierung vorgestellt. Auf der Grundlage dieser Modellierungssprache werden Geschäftsprozess-Modelle einfacher Komplexität unter Verwendung eines Werkzeuges entwickelt. Außerdem werden Geschäftsprozess-Modelle im Hinblick auf verschiedene Kriterien wie z. B. Prozesszeit, Prozessqualität oder Prozesskosten bewertet und verbessert.					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur					
Voraussetzungen für die Vergabe von Credits					

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Grundlagen der Automatisierungstechnik

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Grundlagen der Automatisierungstechnik (135110)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Bernd Kuhlenkötter Lehrende: Prof. Dr.-Ing. Bernd Kuhlenkötter					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes)					
Zielsetzung:					
<ul style="list-style-type: none">• Die Studierenden sollen in der Lage sein, aktuelle Entwicklungen und Trends in der Automatisierungstechnik darzulegen sowie Entwicklungsprozesse für automatisierte technische Systeme erläutern und die entsprechenden Entwicklungsmethoden anwenden zu können.• Sie sollen durch Absolvieren des Kurses in die Lage gebracht werden, das Funktionsprinzip und den Hardware-Aufbau einer SPS darzulegen und Automatisierungsaufgaben im Bereich der SPS- und NC-Programmierung mit methodischer Vorgehensweise zu bearbeiten.• Zudem sollen sie die Kenntnisse erlangen, Robotersysteme für den Einsatz in unterschiedlichen Automatisierungsaufgaben kritisch zu bewerten, geeignete Systeme auszuwählen sowie Sicherheitsrisiken der Automatisierungstechnik zu beurteilen.					
Kenntnisse:					
<ul style="list-style-type: none">• Die Studierenden kennen wesentliche Methoden und Verfahren der Ingenieurwissenschaften / des Maschinenbaus, verfügen über entsprechendes Fachvokabular und kennen Anwendungsbeispiele.• Die Studierenden kennen vertiefte ingenieurwissenschaftliche Grundlagen im Bereich ihres Studienschwerpunkts.					
Fertigkeiten:					
<ul style="list-style-type: none">• Die Studierenden verfügen über die Fähigkeit zu vernetztem und kritischem Denken.• Die Studierenden praktizieren erste Ansätze wissenschaftlichen Lernens und Denkens.• Die Studierenden können ingenieurtechnische Probleme modellieren und lösen.• Die Studierenden können komplexe mathematische Problemstellungen in physikalischen Systemen (ggf. fachübergreifend) mit geeigneten Methoden lösen.• Die Studierenden haben die Fähigkeit zu vernetztem und kritischem Denken ausgebaut und sind in der Lage etablierte Methoden und Verfahren auszuwählen und anzuwenden.					
Kompetenzen:					
<ul style="list-style-type: none">• Die Studierenden verfügen über fachübergreifende Methodenkompetenz.					

- Die Studierenden können Erkenntnisse / Fertigkeiten auf konkrete maschinenbauliche /ingenieurwissenschaftliche Problemstellungen übertragen.

Inhalt

Die Vorlesung „Grundlagen der Automatisierungstechnik“ (GdA) stellt die Themen der industriellen Automatisierung mit dem Fokus auf der Industrierobotik dar. Grundlegende Anwendungsgebiete, wie der Einsatz von Industrierobotik in Lackierstraßen oder Schweißapplikationen, werden neben der historischen Entwicklung der Automatisierungstechnik aufgezeigt. Ein Schwerpunkt der Vorlesung sind die in Bezug auf Automatisierungsaufgaben häufig genutzten Steuerungen mittels SPS und NC/CNC. Neben der Erläuterung des Hardwareaufbaus und des Funktionsprinzips einer SPS werden in vorlesungsbegleitenden Übungen eigene SPS-Programme erstellt. Innerhalb der Steuerungen spielt die Signalverarbeitung von der Erfassung der Sensorsignale über die Verarbeitung und Ausgabe sowie die Art der Kommunikation der Daten untereinander eine wesentliche Rolle. Weiterhin werden innerhalb der Vorlesung Projektabläufe und Planungen von beispielhaften automatisierten Prozessen mit den Studierenden erarbeitet. Einen weiteren Schwerpunkt der Vorlesungs- und Übungseinheiten bilden die vermittelten Grundlagen zur industriellen Robotik. Dabei wird zunächst die Entwicklung der Industrierobotik dargelegt. Des Weiteren werden die wesentlichen Bestandteile eines Robotersystems gelehrt und verschiedene Industrierobotertypen und deren Einsatzgebiete in der Automatisierungstechnik vorgestellt. Die prinzipielle Funktionsweise von Robotersteuerungen wird in weiteren Vorlesungs- und Übungseinheiten vertieft. Die Lehrveranstaltung schließt mit einer Einführung in die Grundlagen der Kommunikationstechnik, Sensorik und Sicherheitstechnik im Themenfeld der Automatisierung ab. Die Inhalte der Vorlesung bereiten Studierende auf die Arbeit als Automatisierungsingenieur vor. Vorträge von Gastreferenten aus Industrie und Forschung zeigen praxisnahe Anwendungsbeispiele aus der Automatisierungstechnik auf und ergänzen somit die Lehrveranstaltung. Übungen dienen der weiteren Vertiefung des gelesenen Lehrstoffes.

Lehrformen

Vorlesung mit Übung

Prüfungsformen

Klausur (180 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Highlights of Theoretical Computer Science [B.Sc.] Highlights of Theoretical Computer Science [B.Sc.]					
Modul-Nr./Code	Credits 10 CP	Workload 300 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Highlights of Theoretical Computer Science (211057)			Kontaktzeit 90 h	Selbststudium 210	Gruppengröße 40 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen Successful completion of an introductory course on theoretical computer science (covering formal languages, basics of complexity theory including NP-completeness and reductions, basics of computability theory). Interest and motivation to learn about theoretical concepts.		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Prof. Dr. Thomas Zeume Lehrende: Prof. Dr. Michael Walter Prof. Dr. Thomas Zeume Dr. Vladimir Lysikov					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit / Informationstechnik					
Vorkenntnisse					
Lernziele (learning outcomes) You will know some of the most important results and insights of modern theoretical computer science. You will learn approaches and techniques that go well beyond a first course. You will be able to recognize when these can be used and how to adapt them to new situations. You will be able to independently acquire new knowledge in this area.					
Inhalt The insights and techniques of modern theoretical computer science have been key for advances in all areas of computer science. In this course, we will discuss some highlights and the techniques that underpin them. Possible topics that we might cover: <ul style="list-style-type: none"> • Computational models (is there life beyond Turing machines?) • Kolmogorov complexity (what is the shortest program that produces some output?) • Communication complexity (how many bits must Alice and Bob exchange to jointly solve a problem?) • Fine-grained complexity (are some easy problems easier than others? and why?) • Fast multiplication of numbers and matrices (can you beat the high-school method?) • Randomness (does it really help to compute faster?) • Circuit lower bounds (why is it so hard to prove that problems are hard?) • Convex optimization (how to maximize profit if all you can ask are yes/no questions) • Hardness of approximation (how easy is it to find near-optimal solutions?) • Cryptography and computation 					

If you enjoyed your first course in theoretical computer science in the Bachelor's and would like to deepen your knowledge by getting an overview of the fascinating theory of computing, then this course will be exactly right for you.

Lehrformen

Vorlesung mit Übung

Prüfungsformen

Final module examination. Format will depend on number of participants.

Voraussetzungen für die Vergabe von Credits

Bestandene Modulabschlussprüfung (schriftliche Klausur 180 Minuten / mündliche Prüfung 15-45 Minuten)

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

10/158: B.Sc. Informatik [PO 22]

10/170: B.Sc. Informatik [PO 20]

10/168: B.Sc. Angewandte Informatik [PO 22]

10/170: B.Sc. Angewandte Informatik [PO 20]

10/150: B.Sc. IT-Sicherheit / Informationstechnik [PO 22]

10/149: B.Sc. IT-Sicherheit / Informationstechnik [PO 20]

Titel des Moduls: Human-Computer Interaction [B.Sc.] Human-Computer Interaction [B.Sc.]					
Modul-Nr./Code	Credits 6 CP	Workload 180h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Human-Computer Interaction (212024)			Kontaktzeit 60h (4 SWS)	Selbststudium 120h	Gruppengröße Studierende
Unterrichtssprache Englisch oder Deutsch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Karola Marky Lehrende: Prof. Dr. Karola Marky					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik B.Sc. IT-Sicherheit					
Vorkenntnisse Keine					
Lernziele (learning outcomes) Die Vorlesung führt in grundlegende Konzepte, Modelle und Theorien aus dem Bereich der Mensch-Computer-Interaktion (HCI) ein. Nach der Teilnahme an der Vorlesung können die Studierenden <ul style="list-style-type: none"> • verstehen die psychologischen Grundlagen der Gestaltung von Benutzeroberflächen • sind mit Methoden des nutzerzentrierten Gestaltungsprozesses vertraut • haben einen Überblick über gängige UI-Konzepte erhalten • haben Evaluationstechniken erlernt und angewendet • haben Erfahrungen im Prototyping neuartiger Interfaces gesammelt (z.B. durch 3D-Druck) 					
Inhalt <ul style="list-style-type: none"> • Theoretische Grundlagen aus Psychologie und Interaktionsdesign als Basis für die Gestaltung von Benutzerschnittstellen • Überblick über verschiedene Arten von traditionellen Benutzerschnittstellen (z.B. Kommandozeilenschnittstellen, grafische Benutzerschnittstellen) • Überblick über fortgeschrittene Benutzerschnittstellen (z. B. mobile Schnittstellen, stiftbasierte Schnittstellen, greifbare Benutzerschnittstellen) • Postreale Schnittstellen (z. B. erweiterte und virtuelle Realität) • Einführung in Benutzerstudien: quantitative Evaluationsmethoden inkl. Grundlagen der statistischen Auswertung; qualitative Evaluationsmethoden • Nutzerzentrierte Softwareentwicklung • Interface-Prototyping mit modernen Technologien (z.B. 3D-Druck, Laserschneiden) 					
Lehrformen <ul style="list-style-type: none"> • Vorlesung • Praktische Übung inkl. Entwicklung eines Interface-Prototyps und Evaluierung 					
Prüfungsformen Klausur (90 Minuten)					

Voraussetzungen für die Vergabe von Credits

Bestehen der Klausur und erfolgreiche Teilnahme an den Übungen.
Bonuspunkte für die erfolgreiche Teilnahme an den Übungen.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

6/158: B.Sc. Informatik

6/168: B.Sc. Angewandte Informatik

6/150: B.Sc. IT-Sicherheit

Titel des Moduls: Inklusives IT-Design (Kein Angebot im SS 25)
Inclusive IT design (no offer in SS 25)

Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Inklusives IT-Design (260020)			Kontaktzeit 60 h	Selbststudium 120 h	Gruppengröße 20 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen Keine		

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr. Christian Meske
 Lehrende: Prof. Dr. Christian Meske
 Markus Jelonek, M.Sc.

Verwendung des Moduls

B.Sc. Angewandte Informatik

M.Sc. Angewandte Informatik [nur im SS 23 als Vertiefungsmodul, ansonsten als Freies Wahlmodul]

Vorkenntnisse

Lernziele (learning outcomes)

Die Veranstaltung richtet sich an Studierende mit Interesse an Themen der Inklusion und dem Nutzerzentrierten Design.

Die Studierenden

- ... kennen die Perspektiven von Endnutzer:innen mit geistiger und/oder körperlicher Behinderung hinsichtlich Interaktionsdesign, Usability und User Experience und können diese bei der Entwicklung eines Prototyps anwenden, begründen und beurteilen.
- ... leisten einen Dienst an der Gesellschaft, indem sie reale Probleme bzgl. der Inklusion von Menschen mit Behinderungen im Rahmen der digitalen Transformation aufgreifen, analysieren, beurteilen und gemeinsam prototypische Lösungen konstruieren (Service Learning).
- ... können partizipative und Design Science Methoden mit Aspekten der Inklusion und des Wohlbefindens in zukünftigen IT-Design-Prozessen zusammenführen und beurteilen.
- ... können theoretisch-konzeptionelle Design-Science-Vorgehensmodelle mit einer praxisorientierten Anwendung im Themenfeld des Positive Computing anwenden.
- ... kennen Anforderungen an inklusives IT-Design und können diese praktisch anwenden.
- ... können Fachwissen, welches in anderen Fächern erworben wurde (UX, Software-Ergonomie und Usability Engineering, Positive Computing, etc.), problemorientiert anwenden und beurteilen sowie neues Wissen selbstständig aneignen.
- ... können zielgerichtet und interdisziplinär zusammenarbeiten sowie Lösungen in einem festen Zeitraum eigenständig konstruieren, analysieren und präsentieren.
- ... können die Vorteile und Herausforderungen bei der Entwicklung von Anwendungen mit und für die spezifische Zielgruppe beschreiben.

- ... können Aspekte des agilen IT-Projektmanagements für Ihren Praxisteil anwenden.
- ... können Aspekte des Positive Computing in ihrer Prototypenwicklung anwenden.

Inhalt

- Inclusive und accessible IT-Design
- Universal Design vs. Inclusive and Accessible Design
- Barrierefreiheit
- UX und Usability in inclusive und accessible IT-Design
- Design Science Research
- Partizipative Methoden und Co-Creation

Lehrformen

- Vorlesung (Flipped Classroom, hybrid) mit praktischen Phasen in Kooperation mit Studierenden der Hochschule Ruhr-West (digital und analog)
- Praxisorientierte Blockveranstaltungen und Projektphasen mit Praxispartner vor Ort und digital

Prüfungsformen

Projektarbeit

Voraussetzungen für die Vergabe von Credits

- Regelmäßige Teilnahme an Pflichtterminen
- Bestandene Zwischen- und Abschlusspräsentation
- Bestandene Abgabe der Dokumentation und Präsentation

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

6/168: B.Sc. Angewandte Informatik [PO22]

6/170: B.Sc. Angewandte Informatik [PO20]

Titel des Moduls: Introduction to Computational Neuroscience Introduction to Computational Neuroscience					
Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Introduction to Computational Neuroscience (211046)			Kontaktzeit 60 h	Selbststudium 120 h	Gruppengröße 50 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Sen Cheng Lehrende: Prof. Dr. Sen Cheng					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse essentiell: (Higher) Mathematics 1 und 2, Statistics, Programming Neuroscience (not required, but nice to have)					
Lernziele (learning outcomes)					
Inhalt					
Lehrformen Vorlesung mit Übung					
Prüfungsformen					
Voraussetzungen für die Vergabe von Credits					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Introduction to Data Science Introduction to Data Science					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Introduction to Data Science (212039)			Kontaktzeit 60h (4 SWS)	Selbststudium 90 h	Gruppengröße 100 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen Keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Robert Schmidt Prof. Dr. Bilal Zafar Lehrende: Prof. Dr. Robert Schmidt Prof. Dr. Bilal Zafar					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik					
Vorkenntnisse Grundkenntnisse in Infinitesimalrechnung, linearer Algebra und Programmierung in Python					
Lernziele (learning outcomes) Am Ende dieses Kurses werden Sie mit folgenden Themen vertraut sein <ol style="list-style-type: none"> 1. Die wichtigsten modernen Methoden für datengestützte Vorhersagen 2. Methoden zur Verarbeitung, Erkundung und Visualisierung von Daten verschiedener Modalitäten wie Bild, Text und Tabellen 3. Aufbau von Proof-of-Concept-Code für die Lösung realer Data-Science-Probleme 4. Fragen rund um Vertrauen und mögliche Abhilfemaßnahmen bei Anwendungen von Data Science. 					
Inhalt Data Science ist ein sich schnell entwickelnder Bereich mit zahlreichen Anwendungsgebieten. In diesem Kurs lernen Sie grundlegende Werkzeuge von Data Science kennen. Sie werden auch mit fortgeschrittenen Methoden des Deep Learning und deren praktischen Anwendungen vertraut gemacht. Im ersten Teil des Kurses erhalten Sie eine Einführung in die grundlegenden statistischen Methoden, die Data Science zugrunde liegen. Sie erlernen auch Techniken zur Analyse und Visualisierung von Datensätzen unterschiedlicher Modalitäten wie Text, Bilder und Tabellen. Sie werden tief in die datengesteuerten Vorhersagemethoden des maschinellen Lernens und des Deep Learning eintauchen. Im letzten Teil des Kurses führen wir Sie in fortgeschrittene Themen ein, einschließlich der jüngsten Fortschritte bei der Modellierung großer Sprachen und der Nutzung datengesteuerter Entscheidungsfindung auf vertrauenswürdige Weise.					
Lehrformen Jede Sitzung des Kurses besteht aus einem Vorlesungsteil, in dem die theoretischen Konzepte vorgestellt werden, und einem praktischen Teil, in dem Sie praktische Erfahrungen mit Jupyter Notebooks sammeln können.					
Prüfungsformen Kombinierte Prüfung: Klausur 120 minutes (70%) + Aufgaben (30%) (beide Teile müssen bestanden werden)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur und bestandene Aufgaben					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)					

5/168: B.Sc. Angewandte Informatik

5/158: B.Sc. Informatik

Titel des Moduls: Künstliche Neuronale Netze
Artificial Neural Networks

Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Artificial Neural Networks (212006)			Kontaktzeit 60 h	Selbststudium 120 h	Gruppengröße 150 Studierende

Unterrichtssprache Englisch	Teilnahmevoraussetzungen Keine
---------------------------------------	--

Modulbeauftragte/r und hauptamtlich Lehrende
 Modulbeauftragte/r: Prof. Dr. Sen Cheng
 Lehrende: Prof. Dr. Sen Cheng

Verwendung des Moduls
 B.Sc. Angewandte Informatik

 B.Sc. Informatik

Vorkenntnisse
 Grundkenntnisse in der Infinitesimalrechnung, linearen Algebra, Statistik und Informatik. Erfahrung mit einer höheren Programmiersprache.

Lernziele (learning outcomes)
 Die mathematischen Grundlagen, Möglichkeiten und Beschränkungen überwachter Lernverfahren für Regression und Klassifikation mit künstlichen neuronalen Netzen (KNN), sowie für deren Anwendung erforderliche praktische Kenntnisse werden vermittelt.

Nach dem erfolgreichen Abschluss des Moduls

- verstehen Studierende die theoretisch-mathematischen Grundlagen von KNN im Kontext des überwachten Lernens.
- können Studierende selbstständig zwischen verschiedenen KNN unterscheiden und in einer Anwendungssituation das geeignete Verfahren auswählen.
- können Studierende grundlegende Verfahren selbstständig in einer höheren Programmiersprache implementieren, sowie ihre eigene Implementierung und Standard- Implementierung anderer auf Daten anwenden.
- können Studierende Ergebnis der KNN selbstständig interpretieren, insbesondere erkennen, wann sie unrealistisch sind.

Inhalt
 Verfahren: Struktur von Optimierungsproblemen, Regression, logistische Regression, biologische neuronal Netze, Modellselektion, universelle Approximationstheorem, Perzeptron, mehr-schichtiges Perzeptron, Backpropagation, tiefe neuronale Netze, rekurrente neuronale Netze, Long-Short Term Memory, Hopfield Netze, Boltzmann-Machine
 Software: python, numpy, matplotlib, scikit-learn, tensorflow

Lehrformen
 Vorlesung, Hausaufgaben, angeleitete Übungen am Computer

Prüfungsformen
 Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

6/168: B.Sc. Angewandte Informatik [PO 22]

6/158: B.Sc. Informatik [PO 22]

Titel des Moduls: Mathematics for Modeling and Data Analysis
Mathematics for Modeling and Data Analysis

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Mathematics for Modeling and Data Analysis (211047)			Kontaktzeit 4 SWS (60 h)	Selbststudium 90 h	Gruppengröße 30 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott					
Verwendung des Moduls B.Sc. Angewandte Informatik M.Sc. Computer Science					
Vorkenntnisse Grundkenntnisse in Infinitesimalrechnung und linearer Algebra.					
Lernziele (learning outcomes) Nach erfolgreichem Abschluss dieses Kurses <ul style="list-style-type: none"> • kennen die Teilnehmenden das in diesem Kurs behandelte Material, siehe Inhalt, • haben sie ein intuitives Verständnis der grundlegenden Konzepte und können damit arbeiten, • können sie sich über all dies auf Englisch verständigen. 					
Inhalt Dieser Kurs behandelt mathematische Methoden, die für die Modellierung und Datenanalyse relevant sind. Besonderer Wert wird auf ein intuitives Verständnis gelegt, das für eine kreative Beherrschung der Mathematik erforderlich ist. Die folgenden Themen werden behandelt: <ul style="list-style-type: none"> • Funktionen und deren Visualisierung • Vektorräume • Matrizen als Transformationen • Systeme von linearen Differentialgleichungen • Qualitative Analyse von nichtlinearen Differentialgleichungen • Bayessche Theorie • Markov-Ketten 					
Lehrformen Dieser Kurs wird nach dem Konzept des "flipped/inverted classroom" durchgeführt. Zunächst arbeiten die Studierenden das Online-Material selbständig durch. In der Vorlesungszeit diskutieren wir dann den Stoff, finden Verbindungen zu anderen Themen, stellen Fragen und versuchen, sie zu beantworten. Im Tutorium wird das neu erworbene Wissen in analytischen Übungen angewendet und dadurch vertieft. Ich ermutige alle Studierenden, sowohl in der Selbstlernzeit als auch im Tutorium in Teams zu arbeiten.					
Prüfungsformen Mündliche Modulabschlussprüfung 15-45 Minuten.					
Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung.					

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik

5/97: M.Sc. Computer Science

Titel des Moduls: Menschenzentrierte Robotik					
Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Menschenzentrierte Robotik (136070)			Kontaktzeit 30 h	Selbststudium 150 h	Gruppengröße 35 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Bernd Kuhlenkötter Lehrende: Prof. Dr. Annette Kluge, Prof. Dr.-Ing. Bernd Kuhlenkötter, Jun.-Prof. Dr. Laura Kunold					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Für den Kurs sollten die Studierenden Teamfähigkeit mitbringen und Interesse an interdisziplinären Themen haben, die über den ingenieurwissenschaftlichen Bereich hinausgehen (wie z.B. die psychologischen Implikationen der Robotik).					
Lernziele (learning outcomes) Zielsetzung: <ul style="list-style-type: none"> • Alle Studierenden haben Grundkenntnisse über die Entwicklungen, Anwendungsbereiche und die aktuellen Trends im Bereich der Mobilen und Servicerobotik. • Die Studierenden der Ingenieurwissenschaften verstehen die Funktionsweise und den Aufbau von Robotersystemen und sind in der Lage diese zu programmieren • Sie haben Grundkenntnisse über Forschungsmethoden der Mensch-Roboter-Interaktion, Mensch-Roboter-Kollaboration und sind in der Lage Gestaltungsempfehlungen auf Basis empirischer Befunde abzuleiten. • Alle Studierenden sind in der Lage die multimediale Landschaft zur Kommunikation zwischen Roboter und Mensch differenziert zu betrachten. • Sie haben die Fähigkeit kleinere Projekte selbständig innerhalb einer Gruppe zu planen und durchzuführen. Kompetenzen: <ul style="list-style-type: none"> • Die Studierenden verfügen über fachübergreifende Methodenkompetenz. • Sie erlernen die Arbeit in interdisziplinären Teams. • Sie können Erkenntnisse/Fertigkeiten auf konkrete maschinenbauliche/ingenieurwissenschaftliche Problemstellungen übertragen. • Sie lernen Grundlagen anderer Disziplinen im interdisziplinären Austausch kennen. • Sie verfügen über experimentelles Fachvokabular, kennen empirische Forschungsmethoden und die Grundlagen der Sozialpsychologie. • Sie können disziplinübergreifende Inhalte aus nicht technischen Bereichen reflektieren und verantwortungsbewusst neue Ansätze in den Projektarbeiten entwickeln. • Sie können durch die Projektarbeiten effektiv und effizient in Teams kommunizieren, diskutieren und ihre Arbeiten im Anschluss präsentieren 					
Inhalt Die Studierenden werden in fachübergreifenden Gruppen an konkreten Problemstellungen im Bereich der menschenzentrierten Robotik arbeiten. Zur Gestaltung des sozio-technischen Systems aus Mensch(en) und Roboter(n), werden sowohl ingenieurwissenschaftliche als auch psychologische Fragen berücksichtigt.					

Vorbereitend hierfür wird aus technischer Sicht eine thematische Einführung in die Historie, Anwendungsfeldern und Funktionsweisen von Robotersystemen gegeben. Dabei wird vor allem auf die mobile Servicerobotik und die Mensch-Roboter-Kollaboration eingegangen. Zur menschengerechten Gestaltung der Interaktion mit der Roboterplattform, wird eine Einführung in psychologische Effekte der Mensch-Technik-Interaktion gegeben, sowie die soziale Robotik und ihre Anwendungsfelder vorgestellt.

Auf dieser Basis bearbeiten die Studierenden dann in interdisziplinären Gruppen individuelle Problemstellungen unter wissenschaftlichen Gesichtspunkten zur Weiterentwicklung einer mobilen Roboterplattform. Hierbei steht thematisch die erfolgreiche Kommunikation und Interaktion zwischen Mensch und Roboter (und der Eindruck des Roboters auf dem Menschen) im Vordergrund. Es besteht die Möglichkeit die ausgearbeiteten Lösungsansätze zu implementieren und somit eine Validierung des Konzeptes durchzuführen.

Die Studierenden werden bei der Projektplanung und dem Projektmanagement unterstützt, indem ihnen die Grundlagen des Projektablaufs für die jeweiligen Projektphasen vermittelt werden. Während der Projektlaufzeit wird durch Zwischengespräche die Projektentwicklung überprüft und reguliert. So werden die in der Lehrveranstaltung vorgestellten Methoden und das erlernte Wissen praktisch angewendet und das Arbeiten in interdisziplinären Projektgruppen eingeübt.

Lehrformen

Vorlesung (2 SWS)

Prüfungsformen

Hausarbeit (semesterbegleitend)

Voraussetzungen für die Vergabe von Credits

- Bestandene Modulabschlussprüfung: Dokumentation und Präsentation (Hausarbeit)
- Teilnahme an allen Zwischengesprächen

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

6/168: B.Sc. Angewandte Informatik [PO 22]

6/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Methoden der Bioinformatik					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Sommersemester	Dauer 1 Semester
Lehrveranstaltungen Methoden der Bioinformatik: Vorlesung (190502) Methoden der Bioinformatik: Übung (190522)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Axel Mosig Lehrende: Prof. Dr. Axel Mosig					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Grundkenntnisse in Biologie					
Lernziele (learning outcomes) Die Studierenden erlernen interdisziplinäre Denkweisen und notwendige Grundkenntnisse, um aktuelle Forschungsthemen der Bioinformatik verfolgen zu können (Vorlesung). Anhand von Fragestellungen der Biologie werden Fähigkeiten des algorithmischen und statistischen Modellierens und Problemlösens erworben (Übungsaufgaben).					
Inhalt <ul style="list-style-type: none"> • Algorithmische und statistische Grundlagen der Bioinformatik (Reguläre Ausdrücke, Endliche Automaten, Turing Maschinen, Komplexität, Dynamische Programmierung, Maximum Likelihood, Hidden Markov Modelle, Poisson Prozesse) • Algorithmen zur Analyse von Sequenz und Struktur von Bio-Molekülen, Rekonstruktion evolutionärer Beziehungen zwischen Sequenzen und Strukturen 					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur (120 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Natural language processing with deep learning [B.Sc]
Natural language processing with deep learning [B.Sc]

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen			Kontaktzeit 4 SWS (60 h)	Selbststudium 90 h	Gruppengröße 100 Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen Eigenes Laptop mit Python und einer geeigneten IDE für die Übungen mitbringen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Ivan Habernal Lehrende: Prof. Dr. Ivan Habernal					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik					
Vorkenntnisse Praktisch: Kenntnisse in Python (Projekt-Strukturen, OOP, Unit-Tests), Kenntnisse in Numpy sind von Vorteil Theoretisch: Grundlagen der Analysis (Ableitungen) und der linearen Algebra (Vektoren), obwohl wir das wiederholen werden. Einige Grundkenntnisse der Linguistik (z. B. Syntax) auf Oberschulniveau sind ebenfalls von Vorteil.					
Lernziele (learning outcomes) Die Studierenden <ul style="list-style-type: none"> • verstehen das Rückgrat des modernen NLP, wie z.B. Embeddings und Transformer-basierte Modelle • können die Fähigkeiten und Schwächen verschiedener Deep-Learning-Modelle im NLP kritisch beurteilen • verstehen verschiedene Tasks im NLP, deren Bewertung und Modellierungsannahmen • können verschiedene Modelle und Ansätze in Python implementieren und sammeln praktische Erfahrungen 					
Inhalt Typische Aufgaben und Datensätze der natürlichen Sprachverarbeitung (NLP) und ihre Bewertung. Warum ist NLP schwierig? Auffrischung der mathematischen Grundlagen, Calculus, gradientenbasierte Optimierung, Backpropagation für beliebige Funktionen. Log-lineare Modelle und Textklassifikation. Tiefe neuronale Netze. Sprachmodelle und Word embeddings. Lernen statischer Word embeddings. Rekurrente neuronale Netze. Encoder-Decoder, Text Generation, Attention und autoregressive Modelle. Transformers. Self-Attention und BERT. Reine Decoder-Modelle und GPT. LLMs: Prompting und In-context Lernen					
Lehrformen Wir werden Vorlesungen und Übungen haben. Die Vorlesungen werden ziemlich interaktiv sein, da versucht wird, die Studierenden in Fragen und kritisches Denken einzubeziehen. Die Folien für jede Vorlesung werden im Voraus hochgeladen, damit die Studierenden sie für ihre Notizen verwenden können. Jede Vorlesung enthält Links zu relevanten Forschungsarbeiten für diejenigen, die sich eingehender mit dem Thema befassen möchten. Die Vorlesungen sind überwiegend theoretisch, d. h. wir behandeln die wichtigsten Konzepte, Ideen und mathematischen Beschreibungen, aber nicht, wie man es in einem bestimmten Framework programmiert. In den praktischen Kursen werden wir genau das Gegenteil tun. Die Studierenden werden mit aktuellen Mainstream-Frameworks für Deep Learning in NLP, wie Pytorch oder Huggingface, experimentieren. Wir werden ein breites Spektrum an Komplexitäten abdecken, von der Programmierung eines einfachen neuronalen Netzes von Grund auf bis hin zur Verwendung eines vortrainierten Sprachmodells.					
Prüfungsformen Klausur (90 Minuten)					

Voraussetzungen für die Vergabe von Credits

Bestandene Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/168: B.Sc. Angewandte Informatik

5/158: B.Sc. Informatik

Titel des Moduls: Numerische Mathematik					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Numerischer Mathematik: Vorlesung (150106) Numerischer Mathematik: Übung (150107)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Markus Reineke Lehrende: Dr. Mario Lipinski					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Inhalte aus Höhere Mathematik 1 und 2					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende die wichtigsten Methoden der Ingenieurmathematik • können Studierende mathematische Problemstellungen in physikalischen Systemen erkennen und lösen • praktizieren Studierende erste Ansätze wissenschaftlichen Lernens und Denkens • verfügen Studierende über fachübergreifende Methodenkompetenz 					
Inhalt Grundlegende Methoden der numerischen Mathematik: Verfahren zur Lösung großer linearer Gleichungssysteme (Gauß-Verfahren, L-R-Zerlegung, Cholesky-Verfahren und Verwandte), Verfahren zur Lösung nichtlinearer Gleichungen und Gleichungssysteme, insbes. Newton-Verfahren mit Modifikationen, Verfahren zur Berechnung von Eigenwerten und Eigenvektoren, Lagrange-, Hermite- und Spline-Interpolation, Verfahren zur numerischen Integration, Numerische Lösung gewöhnlicher Differentialgleichungen, Anfangswertprobleme (Einschrittverfahren, insbes. Runge-Kutta-Verfahren, Ordnung und Konvergenz, Bedeutung der Stabilität und Anwendung auf steife Systeme, Schrittweitenkontrolle, Mehrschrittverfahren).					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur (180 Min)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Programming for Modern Machine Learning Programming for Modern Machine Learning					
Modul-Nr./Code	Credits 6 CP	Workload 180 h	Semester see examination regulations	Turnus Jedes Semester	Dauer 1 Semester
Lehrveranstaltungen Programming for Modern Machine Learning (212040)			Kontaktzeit 60h (4 SWS)	Selbststudium 120 h	Gruppengröße 50 Studierende
Unterrichtssprache English			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Nils Jansen Prof. Dr. Bilal Zafar Lehrende: Prof. Dr. Nils Jansen (http://nilsjansen.org) Prof. Dr. Bilal Zafar (https://informatik.rub.de/zafar)					
Verwendung des Moduls B.Sc. Angewandte Informatik B.Sc. Informatik					
Vorkenntnisse Basic knowledge of statistics and programming.					
Lernziele (learning outcomes) At the end of this course, you will be able to: <ul style="list-style-type: none"> Map a real world problem on a AI/ML paradigm like classification and reinforcement learning. Select the right model type for your task, e.g., XGBoost, MLP, CNN, Transformers. Select the right frameworks (scikit-learn, PyTorch, HuggingFace Transformers)and APIs (OpenAI API) for your tasks. Put together common components like data loaders and training loops to construct your projects. Use best practices like testing and linting to build maintainable code 					
Inhalt Past few years have seen confluence of two related trends: 1/ A rapid adoption of Artificial Intelligence (AI) and Machine Learning (ML) in a wide range of real-world applications across a variety of domains, e.g., healthcare, engineering. 2/ Development of specialized tooling and design patterns for AI/ML workloads. The goal of this course is to introduce the students to various AI/ML prediction paradigms, popular frameworks and design patterns. Specifically, we will build code bases involving (shallow) classification / regression models, CNNs and Transformers using frameworks like scikit-learn, PyTorch and Transformers. We will learn about using data loaders to manage large scale dataset and using GPUs to speed up deep learning workloads. We will also learn about best practices like testing and reproducibility.					
Lehrformen The course will be split into four blocks, each corresponding to a distinct paradigm of machine learning (traditional supervised learning, computer vision, language modeling and reinforcement learning). Each block will have corresponding in-person lectures and mini-projects.					
Prüfungsformen 50% e-Klausur (90 Minuten) + 50% Projektarbeit					
Voraussetzungen für die Vergabe von Credits Passing grade in the practical project and the final exam.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)					

6/168: B.Sc. Angewandte Informatik

6/158: B.Sc. Informatik

Titel des Moduls: Studienprojekt					
Modul-Nr./Code	Credits 8 CP	Workload 240 h	Semester 5	Turnus Jedes Semester	Dauer 1 Semester
Lehrveranstaltungen			Kontaktzeit 30 h	Selbststudium 210 h	Gruppengröße 5 Studierende
Unterrichtssprache Deutsch oder Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: Dozierende im Studiengang Angewandte Informatik					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Gute Programmierkenntnisse (aus den Modulen Informatik 1 und 2, Software-Engineering) und ggf. weitere Kompetenzen je nach Wahl des Projektes					
Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls: <ul style="list-style-type: none"> • können Studierende theoretischen Erkenntnisse zu Softwareentwicklungsprozessen, Projektabläufen und -Phasen an eigenen Projekten anwenden • haben Studierende Ihre Fachkompetenz gemäß der jeweiligen projektspezifischen Aufgabenstellung erweitert • haben Studierende Ihre Fähigkeit zur Teamarbeit verbessert • können Studierende Ihre Projektergebnisse sowohl mündlich als auch schriftlich darstellen und reflektieren 					
Inhalt Im Rahmen des Studienprojekts soll eine Aufgabe aus Bereichen der Angewandten Informatik in Teamarbeit unter Anleitung eines Betreuers gelöst werden. Die angebotenen Projekte decken dabei thematisch die gesamte Bandbreite der Vertiefungsmodule ab. Darüber hinaus werden auch interdisziplinäre Studienprojekte angeboten. In diesen arbeiten z.B. Angewandte Informatiker mit Studierenden der Sozialwissenschaften oder der Sportwissenschaft zusammen.					
Lehrformen Betreutes Projekt					
Prüfungsformen Projektarbeit mit Abschlussbericht					
Voraussetzungen für die Vergabe von Credits Bestandene Projektarbeit und fristgerechte Abgabe des Abschlussberichts					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 8/168: B.Sc. Angewandte Informatik					

Titel des Moduls: System Performance Evaluation
System Performance Evaluation

Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung / see examination regulations	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen System Performance Evaluation (212033)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Steffen Bondorf Lehrende: Prof. Dr.-Ing. Steffen Bondorf					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik					
Vorkenntnisse Programmierkenntnisse in Java und Python					
Lernziele (learning outcomes) Die Studierenden sollen eine breite Kenntnis über folgende Themenfelder erlangen: <ul style="list-style-type: none"> • Systemmodellierung und -leistungsanalyse • Entwurf von Experimenten • Messungen • Simulation 					
Inhalt Die Veranstaltung „System Performance Evaluation“ vermittelt grundlegende Techniken für Modellierung von Computersystemen und der Quantifizierung ihrer Leistung. Im Vordergrund stehen der Entwurf von Experimenten, Simulation und Warteschlangen-theorie.					
Lehrformen Die Vorlesung wird als seminaristischer Unterricht abgehalten, die praktischen Übungen am Rechner können weitere Lehrformen wie Gruppen- und Projektarbeit beinhalten.					
Prüfungsformen 50% semesterbegleitendes Projekt + 50% Abschlussprüfung (geplant Klausur, 60 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandenes Projekt und bestandene Modulabschlussprüfung					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/158: B.Sc. Informatik [PO 22] 5/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Technische Informatik 3
Computer engineering 3

Modul-Nr./Code	Credits 5 CP	Workload 150 Stunden	Semester 3	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Technische Informatik 3 - Hardwareprogrammierung (212003)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Englisch			Teilnahmevoraussetzungen keine		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Tim Güneysu Lehrende: Prof. Dr.-Ing. Tim Güneysu					
Verwendung des Moduls B.Sc. Informatik B.Sc. Angewandte Informatik					
Vorkenntnisse Inhalte der Module Informatik 1; Programmierung, Informatik 2; Algorithmen und Daten-strukturen, Technische Informatik 1; Rechnerarchitektur und Technische Informatik 2; Digitaltechnik					
Lernziele (learning outcomes) Die Studierenden sollen Kenntnisse über technische Herausforderungen bei der anwendungsbezogenen Entwicklung von eingebetteten Systemen sowie des Internet of Things erlernen. Im Vordergrund der Veranstaltung stehen die maschinennahe Programmierung sowie die problemgerechte Integration von Aktorik und Sensorik. Nach dem erfolgreichen Abschluss des Moduls - kennen Studierende das Entwicklungs- und Programmiermodell sowie spezifische Eigenschaften von Mikrocontrollern als zentrale Grundlage eines eingebetteten Systems - haben Studierende die Fähigkeit zur maschinennahen Programmierung eines ausgewählten Mikrocontrollers zur Realisierung grundlegender Steuerprozesse sowie für die umgebende Peripherie 43 - erlernen Studierende Kommunikations- und Interaktionskonzepte (z.B. serielle Kommunikation via UART, SPI, PCIe etc.) mit externen Komponenten - können Studierende nebenläufige Prozesse strukturieren, in Systemen integrieren sowie damit verbundene potenzielle Probleme verstehen - sind Studierende in der Lage, komplexe anwendungsnahe Anforderungen mittels ausgewählter Hardwarekomponenten in ein eingebettetes System zu realisieren					
Inhalt Die Entwicklung von komplexen eingebetteten Systemen in Form einer zentralen Steuereinheit mit unterschiedlicher Sensorik und Aktorik spielen in vielen Anwendungen eine zentrale Rolle. In der Veranstaltung „Technische Informatik 3“ werden die vielfältigen Aufgabentypen und Realisierungsmöglichkeiten eines eingebetteten Systems sowie dessen anwendungsnaher Entwurf und Implementierung behandelt. Ein besonderes Gewicht in der Veranstaltung wird dabei auf ein anwendungsnahes Beispiel aus der Robotik gelegt, bei der viele der genannten Eigenschaften, Programme und Peripherie entwickelt und gesamtheitlich integriert werden müssen, um die Fähigkeiten eines einfachen Robotersystems auf vorhandener Hardware nachzubilden.					
Lehrformen Vorlesung (als Folien und Tafelvortrag) und Übungen, bei denen die vorgestellten Konzepte und Techniken praktisch an unterschiedlichen Hardware-Architekturen umgesetzt werden. Die Übungen beinhalten Elemente der Gruppen- und Projektarbeit					
Prüfungsformen Klausur (150 Minuten)					
Voraussetzungen für die Vergabe von Credits Bestandene Klausur und erfolgreiche Teilnahme an den Übungen					

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

Titel des Moduls: Vertiefungspraktikum (Angewandte Informatik)					
Modul-Nr./Code	Credits 3 CP	Workload 90 h	Semester 4	Turnus Wintersemester oder Sommersemester	Dauer 1 Semester
Lehrveranstaltungen In jedem Semester wird eine wechselnde Auswahl an Praktika bereitgestellt. Die zugeordneten Veranstaltungen können im Vorlesungsverzeichnis eingesehen werden.			Kontaktzeit Je nach Veranstaltungswahl	Selbststudium Abhängig von Praktikumswahl	Gruppengröße 15 Studierende
Unterrichtssprache Abhängig von Praktikumswahl: Deutsch oder Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Lehrende: siehe jeweiligen Praktikumseintrag					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Gute Programmierkenntnisse, ggf. weitere Vorkenntnisse abhängig vom Praktikum					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • haben Studierende ihre Fähigkeiten im Programmieren in einem Forschungs- oder Anwendungsbereich vertieft und erweitert • können Studierende Softwarebibliotheken zur Lösung forschungsnaher Fragestellungen anwenden • je nach gewähltem Praktikum können noch weitere Lernziele dazu kommen 					
Inhalt Es werden Praktika zu verschiedenen relevanten Themen angeboten, wie z.B. Grundlagen der Roboterprogrammierung, Erklärbare Künstliche Intelligenz - Programmierpraktikum oder Open Neural Data. Weiterführende Informationen zu den jeweiligen Praktika finden Sie im Vorlesungsverzeichnis.					
Lehrformen Praktikum im Block oder als semesterbegleitende Veranstaltung.					
Prüfungsformen Praktikum					
Voraussetzungen für die Vergabe von Credits Erfolgreiche Teilnahme am Praktikum					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) unbenotet					

Titel des Moduls: Vertiefungsseminar (Angewandte Informatik)					
Modul-Nr./Code	Credits 3 CP	Workload 90 h	Semester 6	Turnus Jedes Semester	Dauer 1 Semester
Lehrveranstaltungen In jedem Semester wird eine wechselnde Auswahl an Seminaren bereitgestellt. Die zugeordneten Seminare können im Vorlesungsverzeichnis eingesehen werden.			Kontaktzeit 30 h	Selbststudium 60 h	Gruppengröße 15 Studierende
Unterrichtssprache Je nach Seminarwahl: Deutsch oder Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: siehe jeweilige Seminare					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Die Vertiefungsseminare beziehen sich in der Regel auf Inhalte aus bestimmten Pflicht- oder Vertiefungsmodulen, die im Vorfeld absolviert worden sein sollten.					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • verfügen Studierende über vertiefte wissenschaftliche Kenntnisse in dem ausgewählten Seminarthema • haben Studierende das Halten eines wissenschaftlichen Vortrags praktisch eingeübt und können Forschungsergebnisse eigenständig in einem didaktisch wohl aufbereiteten Vortrag vermitteln • können die Teilnehmer konstruktives Feedback formulieren und entgegennehmen • können Studierende eine schriftliche Ausarbeitung zu ihrem Seminarvortrag verfassen 					
Inhalt Es werden Bachelorseminare zu mehreren relevanten Themen angeboten, wie beispielsweise zu maschinellem Lernen, Algorithmen, theoretischer Informatik oder zu Ingenieurinformatik. Von den angebotenen Themen wählen die Studierenden abhängig von den eigenen Interessen und den individuellen Vertiefungswünschen ein Thema aus. Dieses sollen die Studierenden selbstständig bearbeiten. Dazu gehören die Literaturrecherche, die Einarbeitung in das Thema und schließlich die Präsentation. Nähere Informationen sind zu den jeweiligen Seminaren im Vorlesungsverzeichnis zu entnehmen.					
Lehrformen Seminar					
Prüfungsformen Seminarvortrag und ggf. schriftliche Ausarbeitung.					
Voraussetzungen für die Vergabe von Credits Der Seminarvortrag muss mindestens mit der Note „ausreichend“ bewertet sein. Um die Lernziele zu erreichen, besteht im Seminar Anwesenheitspflicht an mindestens 9 von 10 Terminen. Mehrfaches Fehlen muss durch ein ärztliches Attest entschuldigt werden. Die Anwesenheit beim ersten Termin ist obligatorisch, da zu diesem Termin die Themen verteilt werden. Das Seminar gilt als nicht bestanden, wenn an mehr als einem Termin unentschuldigt gefehlt wurde.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 3/168: B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Virtuelle Produktmodellierung und -visualisierung Virtual Product-Modelling and Visualization					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Virtuelle Produktmodellierung und -visualisierung (135060)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Detlef Gerhard Lehrende: Prof. Dr.-Ing. Detlef Gerhard					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • verfügen Studierende über ein breites, integriertes Wissen über die Herausforderungen moderner Produktentstehungsprozesse und die resultierenden Anforderungen an Softwaresysteme zur Virtuellen Produktmodellierung und -visualisierung. • kennen und verstehen Studierende wesentliche Methoden und Verfahren der Virtuellen Produktmodellierung und -visualisierung einschließlich der angrenzenden Gebiete und der wissenschaftlichen informationstechnischen Grundlagen. Indem sie praktische Beispiele und Aufgaben mit entsprechender Anwendungssoftware bearbeiten, können sie die erlernten Fertigkeiten im Umgang mit Softwaresystemen auf konkrete konstruktionstechnische Problemstellungen übertragen, um diese modellieren und lösen zu können. • haben Studierende ein umfassendes Verständnis vom Zusammenwirken der Softwaresysteme und Produktdatenmodelle innerhalb der verschiedenen Prozessketten in der Produktentstehung und können kritisch die Eignung von Methoden zur Virtuellen Produktmodellierung und -visualisierung zur Konzeption, Konstruktion, Optimierung, Darstellung, Fertigungsvorbereitung und Dokumentation von Produkten differenzieren und beurteilen. • können Studierende Aufgabenstellungen der Virtuellen Produktmodellierung und -visualisierung reflektieren und bewerten sowie selbstgesteuert verfolgen. • können Studierende kooperativ Aufgabenstellungen der Virtuellen Produktmodellierung und -visualisierung in heterogenen Gruppen bearbeiten, Abläufe und Ergebnisse begründen sowie über Sachverhalte umfassend kommunizieren. 					
Inhalt Das Modul vermittelt Methoden und Werkzeuge zur "Virtuellen Produktmodellierung und -visualisierung", insbesondere das dazu erforderliche Grundlagenwissen und die relevanten methodischen Aspekte der systematischen Produktentwicklung. Schwerpunkte bilden dabei die verschiedenen CAD-Modellierungsmethoden (z.B. 3D Flächen und Volumenmodellierung, parametrische Modellierung, Baugruppenmodellierung) entsprechend der Anforderungen aus der Konstruktionsaufgabe sowie die Kombination von Verfahren zur durchgängigen Abbildung von Prozessketten (z.B. für Digital Mockup (DMU), Virtuelle und Augmentierte Realität (VR/AR), Auslegungs- und Nachweisberechnungen, Analyse und Simulation, Additive					

Manufacturing, Produktion (CAM), Digital Factory, Styling, Elektro/Elektronik-CAD) im Produktlebenszyklus mit Aspekten der Integration von Modellen und Werkzeugen.

Lehrformen

Vorlesung mit Übung

Prüfungsformen

Klausur (90 Min., Anteil der Modulnote 100 %)

Studienbegleitende Aufgaben: Hausarbeiten (Sofern die Hausarbeiten vor der Modulabschlussprüfung absolviert werden, sind optional Bonuspunkte für die Klausur möglich) (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben)

Voraussetzungen für die Vergabe von Credits

- Bestandene Klausur
- Bestandene studienbegleitende Aufgaben: Hausarbeiten (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben)

Sowohl die Klausur als auch die Hausarbeit ist über Flexnow anzumelden.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS)

Titel des Moduls: Wirtschaft und Digitalisierung					
Modul-Nr./Code	Credits 5 CP	Workload 150 h	Semester siehe Prüfungsordnung	Turnus Wintersemester	Dauer 1 Semester
Lehrveranstaltungen Wirtschaft und Digitalisierung (076040 und 076041)			Kontaktzeit 60 h	Selbststudium 90 h	Gruppengröße 90 Studierende
Unterrichtssprache Deutsch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Roland Düsing Lehrende: Dr. Roland Düsing					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse					
Lernziele (learning outcomes) Die TeilnehmerInnen des Moduls sollen <ul style="list-style-type: none"> • die Bedeutung des Wirtschaftens erklären können, • die Bestandteile, Funktionen und Eigenschaften eines Markts beschreiben können, • digitale Güter charakterisieren können, • die Besonderheiten der Wertschöpfung auf digitalen Märkten erläutern können, • ausgewählte Marktmodelle darstellen können 					
Inhalt Die Digitalisierung hat Auswirkungen auf zahlreiche Bereiche des wirtschaftlichen Handelns. Digitale Produkte und Dienstleistungen, wie z. B. ePaper oder Cloud-Services, weisen im Vergleich zu materiellen Gütern spezielle Eigenschaften auf, die großen Einfluss auf das Kunden- und Wettbewerber-Verhalten auf digitalen Märkten haben. Die Digitale Transformation in Unternehmen ermöglicht Veränderungen der Wertschöpfung, die zu digitalen Geschäftsmodellen, wie beispielsweise Digitale Plattformen, führen können. In diesem Modul werden die Auswirkungen der Digitalisierung auf das wirtschaftliche Handeln erläutert. Dabei werden zunächst die Grundlagen des Wirtschaftens dargestellt. Im Anschluss daran werden die Konzepte Digitalisierung und Digitale Transformation beschrieben. Schließlich werden die Besonderheiten der Wertschöpfung auf digitalen Märkten veranschaulicht und ausgewählte Marktmodelle skizziert.					
Lehrformen Vorlesung mit Übung					
Prüfungsformen Klausur 60 Minuten					
Voraussetzungen für die Vergabe von Credits Klausur					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20]					

Titel des Moduls: Freie Wahlmodule					
Modul-Nr./Code	Credits 9 CP	Workload 270 h	Semester	Turnus Jedes Semester	Dauer 1 Semester
Lehrveranstaltungen			Kontaktzeit abhängig von Veranstaltungswahl	Selbststudium Je nach Veranstaltungswahl	Gruppengröße Studierende
Unterrichtssprache Je nach Veranstaltungswahl			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studienfachberatung Angewandte Informatik Lehrende:					
Verwendung des Moduls B.Sc. Angewandte Informatik [PO 22]					
Vorkenntnisse abhängig von Veranstaltungswahl					
Lernziele (learning outcomes) Die Teilnehmer erwerben so genannte Schlüsselfähigkeiten in den freien Wahlfächern					
Inhalt Studierende müssen Veranstaltungen im Gesamtumfang von 9 CP absolvieren. Je nach Veranstaltungswahl werden unterschiedliche Inhalte vermittelt. Studierende haben die Möglichkeit unter den Freien Wahlmodulen auch Fächer jenseits der Informatik zu absolvieren und so ihre Soft Skills zu erweitern. Z.B. wird die englische Fachsprache verbessert, in die Grundlagen der Rechtswissenschaften eingeführt oder Grundkenntnisse der Betriebswirtschaft vermittelt. Die Studierenden haben die Möglichkeit eine Auswahl entsprechend der eigenen Interessen zu treffen. Bei der Auswahl geeigneter Lehrveranstaltungen kann die Liste unter https://informatik.rub.de/studium/pruefungsamt/wahlbereich/ hilfreichen Input geben.					
Lehrformen					
Prüfungsformen abhängig von Veranstaltungswahl					
Voraussetzungen für die Vergabe von Credits abhängig von Veranstaltungswahl					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 0/168 (unbenotet): B.Sc. Angewandte Informatik [PO 22]					

Titel des Moduls: Abschlussarbeit (Bachelor AI)					
Modul-Nr./Code	Credits 15 CP	Workload 450 h	Semester 6	Turnus Jedes Semester	Dauer 1 Semester
Lehrveranstaltungen Bachelorarbeit (12 CP) Kolloquium (3 CP)			Kontaktzeit 15 h	Selbststudium 435 h	Gruppengröße Studierende
Unterrichtssprache Deutsch oder Englisch			Teilnahmevoraussetzungen		
Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: Lehrende im Studiengang Angewandte Informatik					
Verwendung des Moduls B.Sc. Angewandte Informatik					
Vorkenntnisse Abhängig von der Themenwahl.					
Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls <ul style="list-style-type: none"> • können Studierende selbstständig und fristgerecht ein wissenschaftliches Thema bearbeiten von der Recherche bis zur Dokumentation der Resultate • können Studierende geeignete wissenschaftliche Verfahren und Methoden, die sie im Studium kennengelernt haben, auswählen und anwenden, um ein konkretes Problem zu lösen • können Studierende ihre Ergebnisse kritisch mit dem Stand der Forschung vergleichen und evaluieren • können Studierende ihre eigenen Ergebnisse angemessen in Wort und Schrift darstellen 					
Inhalt Die Bachelorarbeit ist eine schriftliche Prüfungsarbeit. Es wird die selbstständige Bearbeitung einer anspruchsvollen Aufgabe der Angewandten Informatik unter Anwendung der im Bachelorstudium erworbenen Kenntnisse und Methoden erwartet. Im Anschluss an die Bearbeitung der Bachelorarbeit (12 CP) werden die Ergebnisse in Form eines Kolloquiumsvortrags (3 CP) präsentiert.					
Lehrformen Abschlussarbeit					
Prüfungsformen Schriftliche Bachelorarbeit und Kolloquiumsvortrag					
Voraussetzungen für die Vergabe von Credits Sowohl die Bachelorarbeit als auch der Kolloquiumsvortrag müssen bestanden sein. Die jeweiligen Teilnoten gehen gewichtet nach den CP in die Modulnote ein.					
Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 180 ECTS) 15/168: B.Sc. Angewandte Informatik					