

**Bachelor Studiengang  
Angewandte Informatik**

**PO 13**

**Modulhandbuch**

**Erstellt am: 6. Dezember 2019**

## Wo finde ich Hilfe während des Studium?

### Wichtige Informationsquellen rund um das Studium der Angewandten Informatik:

Studiengangsw Webseite:  
<https://www.ai.rub.de/>

Studienfachberatung Angewandte Informatik:  
<https://www.ai.rub.de/kontakte/studienberatung.html>

Prüfungsamt Angewandte Informatik:  
<https://www.ai.rub.de/kontakte/pruefungsamt.html>

Fachschaftsrat Angewandte Informatik:  
<https://ai-rub.de/>

Bei fachlichen Fragen besteht die Möglichkeit die Dozenten während Ihrer Sprechstunden (siehe individuelle Webseiten) zu kontaktieren.

### Weitere wichtige Kontaktadressen auf dem Campus sind:

Zentrale Studienberatung:  
<https://www.ruhr-uni-bochum.de/zsb/>  
Bietet Hilfe und Coaching bei individuellen Problemen (auch psychologische Betreuung).

Studienfinanzierungsberatung:  
<https://studium.ruhr-uni-bochum.de/de/studienfinanzierung>

Talentscouts der RUB:  
<https://studium.ruhr-uni-bochum.de/de/die-talentscouts-der-rub>  
Beratung zu Stipendien; hierzu werden extra Workshops angeboten.

Beratungszentrum zur Inklusion Behinderter:  
<https://www.akafoe.de/inklusion/>

International Office:  
<http://www.international.rub.de/ausland/index.html.de>  
Beratung zu Studienaufenthalten im Ausland

Wohnheimplätze:  
<https://www.akafoe.de/wohnen/>

### Studiengangsziele:

Ziel des Bachelorstudiums ist ein wissenschaftlich fundiertes, grundlagenorientiertes Studium, das eine breite und in ausgewählten Teilgebieten vertiefte Basis fachlichen Wissens sowie eine umfassende, auch anwendungsorientierte, Methodenkompetenz vermittelt. Die Vermittlung analytischer, kreativer und gestalterischer Fähigkeiten, aber auch anwendungsbezogener Fertigkeiten im Umgang mit modernen Hard- und Softwaresystemen, unter der Maßgabe, innovative informatische Problemlösungskonzepte neu oder weiter zu entwickeln sowie komplexe Anwendungssysteme mit polydisziplinärer Ausrichtung zu schaffen, sind oberstes Ziel des Studiums. Damit eröffnet der Bachelorstudiengang den Berufszugang. Außerdem vermittelt er die wissenschaftlichen Grundlagen für ein Masterstudium.

### Modularisierungskonzept:

Das Studium ist modular aufgebaut. Die Module stellen zeitlich und inhaltlich abgeschlossene Teilqualifikationen dar. Sie haben, abgesehen von wenigen Ausnahmen, einen Workload von mindestens 5 Creditpoints (CP), wobei ein Creditpoint in etwa einer Arbeitsbelastung von 30 Stunden entspricht. In die Berechnung des Workloads fließt neben der Präsenzzeit auch die Zeit für das Selbststudium mit ein (Bearbeitung von Übungsaufgaben, Vor- und Nachbereitung der Vorlesungen, Lesen geeigneter Literatur,?). Während des Studiums müssen Pflichtmodule in Informatik, Mathematik und Betriebswirtschaftslehre erfolgreich abgeschlossen werden. Diese vermitteln wichtiges Grundlagenwissen, welches für das weitere Studium und/oder den späteren Beruf essentiell ist. Der Pflichtbereich macht etwa 2/3 des Studiums aus. In allen anderen Modulen können die Studierenden durch zahlreiche Wahlmöglichkeiten ein individuelles Studienprofil entwickeln.

### Prüfungsformen

Im Bachelorstudiengang sind die folgenden Prüfungsformen vertreten: Klausurarbeiten, Prüfungsgespräche, Studienbegleitende Aufgaben, Projektarbeiten, Seminarbeiträge, Praktika, Bachelorarbeit mit Kolloquiumsbeitrag. Genauer zu den jeweiligen Prüfungsformen kann in der gültigen Prüfungsordnung nachgelesen werden.

## Studienplan Bachelor Angewandte Informatik PO 13

Nr	Modul	Lehrveranstaltungen	Mind. Umfang Modul (LP)	Semester der Modulprüfung	autom. Anmeldung	autom. Wiederanmeldung	Bewertung
<b>Pflichtmodule Mathematik</b>							
1	Höhere Mathematik I	Höhere Mathematik I	9	1	Ja	Ja	benotet
2	Statistik und Wahrscheinlichkeitsrechnung	Statistik und Wahrscheinlichkeitsrechnung	5	1	Ja	Ja	benotet
3	Höhere Mathematik II	Höhere Mathematik II	9	2	Ja	Ja	benotet
4	Diskrete Mathematik I	Diskrete Mathematik I	9	3	Nein	Ja	benotet
<b>Pflichtmodule Informatik</b>							
5	Informatik I	Informatik I	5	1	Ja	Ja	benotet
6	Programmieren in C	Programmieren in C	3	1	Ja	Ja	benotet
7	Computernetze	Computernetze	5	2	Ja	Ja	benotet
8	Informatik II	Informatik II	5	2	Ja	Ja	benotet
9	Objektorientierte Modellierung	Objektorientierte Modellierung	5	2	Ja	Ja	benotet
10	Rechnerarchitektur	Rechnerarchitektur	5	3	Nein	Ja	benotet
11	Theoretische Informatik	Theoretische Informatik	9	3	Nein	Ja	benotet
12	Datenstrukturen	Datenstrukturen	9	4	Nein	Ja	benotet
13	Software-Engineering	Software-Engineering	5	4	Nein	Ja	benotet
14	Web-Engineering	Web-Engineering	5	4	Nein	Ja	benotet
15	Betriebssysteme	Betriebssysteme	5	4	Nein	Ja	benotet
16	Datenschutz	Datenschutz	4	5	Nein	Ja	benotet
17	Datenbanksysteme	Datenbanksysteme	9	5	Nein	Ja	benotet
<b>Pflichtmodule Betriebswirtschaftslehre</b>							
18	Wirtschaftlichkeitsanalyse	Wirtschaftlichkeitsanalyse	5	1	Ja	Ja	benotet
19	Einführung Management Science	Einführung Management Science	6	2	Ja	Ja	benotet
<b>Wahlfächer</b>							
20	Nichttechnische Wahlmodule	Nichttechnische Wahlmodule*	5	1-6	Nein	Nein	unbenotet
<b>Praktische Fächer</b>							
21	Programmierpraktikum	Programmierpraktikum	3	4	Nein	Ja	benotet
22	Studienprojekt	Projektmanagement, Studienprojekt	10	5	Nein	Nein	benotet
<b>Vertiefungsbereich</b>							
23	Vertiefungsmodule	Vertiefungsmodule **	30	3 – 6	Nein	Ja	benotet
24	Vertiefungsseminar	Vertiefungsseminar **	3	6	Nein	Nein	benotet
<b>Bachelorarbeit</b>							
25	Bachelorarbeit und Kolloquium	Bachelorarbeit und Kolloquium	12	6	Nein	Nein	benotet
Summe:			180				

\* Hier können (nahezu) alle Veranstaltungen des Vorlesungsverzeichnisses der RUB, sowie Veranstaltungen im Rahmen der UAMR gewählt werden, sofern es sich dabei um nichttechnische Fächer handelt. Aus der Fakultät Wirtschaftswissenschaften sind nicht alle Veranstaltungen für die AI geöffnet. Eine entsprechende Auflistung befindet sich im aktuellen Modulhandbuch. Eine Liste mit Empfehlungen finden Sie auf der Homepage des Studienganges.

\*\* Die Liste der wählbaren Vertiefungsseminare und Vertiefungsmodule befinden sich im jeweils aktuellen Modulhandbuch, welches vom Gemeinsam beschließenden Ausschuss verabschiedet wird.

**Vertiefungsbereich Bachelor Angewandte Informatik PO13**

Lehrveranstaltung	Einheit	Umfang Modul (LP)	Semester	Autom. Anmeldung	Autom. Wiederanmeldung	Bewertung
<b>Vertiefungsseminare</b>						
Computerlinguistik (Angebot siehe Modulhandbuch)	Philologie	3	WS/SS	Nein	Nein	benotet
Ingenieurinformatik	MB/Baulng	3	WS/SS	Nein	Nein	benotet
Kryptologie und Theoretische Informatik (Angebot siehe Modulhandbuch)	Mathe	3	unregelmäßig	Nein	Nein	benotet
Bachelor-Seminar Maschinelles Lernen	INI	3	SS	Nein	Nein	benotet
<b>Vertiefungsmodule</b>						
<b>Ingenieurinformatik</b>						
Digitaltechnik (Informatik 3)	ETIT	5	WS	Nein	Ja	benotet
Geometrische Modellierung und Visualisierung	Baulng	6	WS	Nein	Ja	benotet
Grundlagen der Automatisierungstechnik	MB	6	WS	Nein	Ja	benotet
Fertigungsautomatisierung	MB	6	SS	Nein	Ja	benotet
Mensch-Maschine-Interaktion (wird im WS 19/20 an Uni-Essen angeboten, siehe Modulhandbuch)	AW	5	WS	Nein	Ja	benotet
Virtuelle Produktentwicklung	MB	6	WS	Nein	Ja	benotet
Umweltinformatik	Baulng	3	WS	Nein	Ja	benotet
<b>Bioinformatik</b>						
Grundlagen der Bioinformatik	Biologie	5	WS	Nein	Ja	benotet
Methoden der Bioinformatik	Biologie	5	SS	Nein	Ja	benotet
<b>Computerlinguistik</b>						
Computerlinguistik 1 (Computerlinguistische Programmierung)	Philologie	5	SS	Nein	Ja	benotet
Einführung in die Linguistik (Einführung in die Syntax; Einführung in die Semantik, Linguistische Methoden)	Philologie	8	WS+SS	Nein	Ja	benotet
Computerlinguistik 2 (Symbolische und statistische Verfahren)	Philologie	5	WS	Nein	Ja	benotet
<b>Kryptologie und theoretische Informatik</b>						
Diskrete Mathematik II	Mathe	6	SS	Nein	Ja	benotet
Zahlentheorie	Mathe	9	SS	Nein	Ja	benotet
Einführung in die Kryptographie I	ETIT	5	WS	Nein	Ja	benotet
Einführung in die Kryptographie II	ETIT	5	SS	Nein	Ja	benotet
Kryptographie auf hardwarebasierten Plattformen	ETIT	5	WS	Nein	Ja	benotet
<b>Neuroinformatik</b>						
Computersehen: Einführung (ehemals Digitale Bildverarbeitung)	INI	5	SS	Nein	Ja	benotet
Künstliche Neuronale Netze	INI	6	WS	Nein	Ja	benotet
Mathematics for Modeling and Data Analysis	INI	6	SS	Nein	Ja	benotet
Mathematical Psychology	INI	6	SS	Nein	Ja	benotet
<b>Programmier- und Simulationstechnik</b>						
Agent-based Modeling in Economics and Business	WIWI	5	WS	Nein	Ja	benotet
Spieleentwicklung in C#	INI	5	WS	Nein	Ja	benotet

## Abkürzungen:

AW: Institut für Arbeitswissenschaften  
 Baulng: Fakultät für Bau- und Umweltingenieurwissenschaften  
 ETIT: Fakultät für Elektrotechnik und Informationstechnik  
 INI: Institut für Neuroinformatik  
 MB: Fakultät für Maschinenbau  
 WIWI: Fakultät für Wirtschaftswissenschaft

SS: Sommersemester  
 WS: Wintersemester

LP: Leistungspunkte (auch Creditpoints oder CP)

# Inhaltsverzeichnis

<b>1</b>	<b>Pflichtmodule Mathematik</b>	<b>1</b>
1.1	Höhere Mathematik I	1
1.2	Statistik und Wahrscheinlichkeitsrechnung	3
1.3	Höhere Mathematik II	5
1.4	Diskrete Mathematik I	7
<b>2</b>	<b>Pflichtmodule Informatik</b>	<b>8</b>
2.1	Informatik I	8
2.2	Programmieren in C	10
2.3	Computernetze	12
2.4	Informatik II	14
2.5	Objektorientierte Modellierung	16
2.6	Rechnerarchitektur	18
2.7	Theoretische Informatik	21
2.8	Datenstrukturen	23
2.9	Software-Engineering	25
2.10	Web-Engineering	27
2.11	Betriebssysteme	29
2.12	Datenschutz	31
2.13	Datenbanksysteme	34
<b>3</b>	<b>Pflichtmodule Betriebswirtschaftlehre</b>	<b>36</b>
3.1	Wirtschaftlichkeitsanalyse	36
3.2	Einführung Management Science	38
<b>4</b>	<b>Nichttechnische Wahlmodule</b>	<b>40</b>
4.1	Nichttechnische Wahlmodule	40
<b>5</b>	<b>Praktische Fächer</b>	<b>42</b>
5.1	Programmierpraktikum	42
5.1.1	Intensivkurs C++	42
5.1.2	Introduction to Python	44
5.1.3	Programmierpraktikum Android	46
5.1.4	Grundlagen Roboterprogrammierung	48
5.2	Studienprojekt	50
5.2.1	Studienprojekt	50
5.2.2	Projektmanagement	52

<b>6</b>	<b>Bachelorarbeit</b>	<b>54</b>
6.1	Bachelorarbeit . . . . .	54
<b>7</b>	<b>Ingenieurinformatik</b>	<b>56</b>
7.1	Digitaltechnik (Informatik 3) . . . . .	56
7.2	Geometrische Modellierung und Visualisierung . . . . .	59
7.3	Grundlagen der Automatisierungstechnik . . . . .	61
7.4	Mensch-Maschine-Interaktion . . . . .	64
7.5	Virtuelle Produktentwicklung . . . . .	67
7.6	Umweltinformatik . . . . .	69
7.7	Fertigungsautomatisierung . . . . .	71
<b>8</b>	<b>Bioinformatik</b>	<b>73</b>
8.1	Grundlagen der Bioinformatik . . . . .	73
8.2	Methoden der Bioinformatik . . . . .	75
<b>9</b>	<b>Computerlinguistik</b>	<b>77</b>
9.1	Computerlinguistik 1 . . . . .	77
9.2	Einführung in die Linguistik . . . . .	79
9.3	Computerlinguistik 2 . . . . .	82
9.4	Symbolische und statistische Verfahren . . . . .	84
<b>10</b>	<b>Kryptologie</b>	<b>86</b>
10.1	Diskrete Mathematik II . . . . .	86
10.2	Zahlentheorie . . . . .	88
10.3	Einführung in die Kryptographie 1 . . . . .	90
10.4	Einführung in die Kryptographie 2 . . . . .	92
10.5	Kryptographie auf hardwarebasierten Plattformen . . . . .	94
<b>11</b>	<b>Neuroinformatik</b>	<b>96</b>
11.1	Computersehen: Einführung . . . . .	96
11.2	Kuenstliche Neuronale Netze . . . . .	98
11.3	Mathematics for Modeling and Data Analysis . . . . .	100
11.4	Mathematical Psychology . . . . .	102
<b>12</b>	<b>Programmier- und Simulationstechnik</b>	<b>104</b>
12.1	Spieleentwicklung in CSharp . . . . .	104
12.2	Agent-based modeling in economics and business . . . . .	106
<b>13</b>	<b>Vertiefungsseminar</b>	<b>108</b>
13.1	Seminar Computerlinguistik . . . . .	108
13.2	Seminar Ingenieurinformatik . . . . .	111
13.3	Seminar Kryptologie und Theoretische Informatik . . . . .	113
13.4	Seminar Maschinelles Lernen . . . . .	115

# Pflichtmodule Mathematik

<b>Höhere Mathematik I</b>					
<b>Modul-Nr:</b> 150160 (Übung: 150161)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 1 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> V: 4 SWS; Ü: 2 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> Vorlesung: Unbegrenzt; Übung: 50 pro Gruppe
<b>Teilnahmevoraussetzungen:</b> Keine <b>Empfohlene Vorkenntnisse:</b>  Mathematische Schulausbildung (Gymnasiale Oberstufe) und die Teilnahme an einem Mathematik-Vorkurs					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls: Die Studierende <ul style="list-style-type: none"> <li>• Verstehen und wenden grundlegende mathematische Begriffe und Notationen an</li> <li>• Nutzen und ausführen die vermittelten mathematischen Methoden</li> <li>• Übertragen und wenden zugehörige Lösungsverfahren auch auf praktische Probleme an</li> </ul> <b>Vermittelte Kompetenzen:</b> Kernkompetenzen: <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> </ul> Fachspezifische Kompetenzen: <ul style="list-style-type: none"> <li>• Mathematik</li> </ul>					
<b>Inhalt:</b> Aussagenlogik, Mengen und Abbildungen, Reelle Zahlen und algebraische Strukturen, Re-					



## KAPITEL 1. PFLICHTMODULE MATHEMATIK

elle Zahlen und algebraische Strukturen, Komplexe Zahlen, Folgen und Reihen, Stetige Funktionen, Differenzialrechnung (in $\mathbb{R}$ ), Integralrechnung (in $\mathbb{R}$ )
<b>Lehrformen:</b> Vortrag der Lehrenden in der Vorlesung (mit zum Teil digitalen Lehrformate), Gruppenarbeit in den Übungen, Ergänzung der Bearbeitung der Hausaufgaben in Einzel- oder Gruppenarbeit durch digitale Aufgaben, online Tests
<b>Prüfungsformen:</b> Klausur (120 Minuten)
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 9 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  PD Dr. Daniela Kacso

<b>Statistik und Wahrscheinlichkeitsrechnung</b>					
<b>Modul-Nr:</b> 070030 (Kolloquium: 070031/070032, Tutorium: 076040)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 1 Sem.	<b>Turnus:</b> immer	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Kolloquium / Tutorium			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> Kolloquium: 60 – 100 Studieren- de, Tutorium: 20 - 40 Studie- rende
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Generische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Abstraktion und logisches Denken (systemische Kompetenz),</li> <li>• Analyse und Strukturierung von Problemen (systemische K.)</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Auswahl geeigneter Methoden zur Problemlösung (instrumentale K.)</li> <li>• Kontextspezifische (ökonomische) Interpretation und Darstellung von Resultaten (kommunikative K.)</li> </ul> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• verfügen Studierende über grundlegende Methodenkenntnisse im Bereich der Wahrscheinlichkeitsrechnung,</li> <li>• kennen Studierende die Konzepte der Schließenden Statistik</li> </ul>					
<p><b>Inhalt:</b></p> <p>Die Veranstaltung besteht aus zwei Teilen. Der erste Teil behandelt die Wahrscheinlichkeitsrechnung. Nach einer kurzen Übersicht über die klassische und die axiomatische Wahrscheinlichkeitsrechnung werden grundlegende Eigenschaften von Zufallsvariablen und für die wirtschaftswissenschaftliche Anwendung wichtigen parametrischen Verteilungen vorgestellt. Anschließend werden Approximationsaussagen, die u.a. auf den Zentralen Grenzwertsätzen beruhen, eingeführt. Im zweiten Teil wird eine Einführung in die Methoden der schließenden Statistik gegeben. Dazu werden die Konzepte von Zufallsstichproben, Punkt- und Intervallschätzung, Signifikanztests und Regressionsanalyse einführend dargestellt. In den vorlesungs-</p>					

## KAPITEL 1. PFLICHTMODULE MATHEMATIK

begleitenden Kolloquien und Tutorien werden die Anwendungen der Methoden anhand von Fallbeispielen und Aufgaben eingeübt.

### **Lehrformen:**

Hörsaalvorlesung mit Medienunterstützung, Kolloquien als seminaristischer Unterricht, Tutorien als seminaristischer Unterricht oder Inverted Classroom, zusätzlich jeweils Selbststudium mit ergänzend bereitgestellten Materialien und Aufgaben

### **Prüfungsformen:**

Modulklausur (90 Minuten)

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

### **Verwendung des Moduls (in anderen Studiengängen):**

Studiengang Management and Economics (Pflichtmodul Grundlagenphase), 2-Fach Bachelor Wirtschaftswissenschaft mit Ausrichtung BWL (Pflichtmodul Grundlagenphase), 2-Fach Bachelor Wirtschaftswissenschaft mit Ausrichtung VWL (Pflichtmodul Grundlagenphase), Nebenfach Wirtschaftswissenschaft, Studierende anderer Fakultäten (über das Zentrum für Ökonomische Bildung)

**Stellenwert der Note für die Endnote:** 5 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Dr. Markus Pape**

Dr. Benno Hildebrandt, Markus Pape

### **Sonstige Informationen:**

Die Modulbezeichnung an der Fakultät für Wirtschaftswissenschaft lautet „Statistik II“. Das Modul „Statistik I“ in den wirtschaftswissenschaftlichen Bachelorstudiengängen beschäftigt sich u.a. mit deskriptiver Statistik, elementarer Zeitreihenanalyse und Indexzahlen.

Die vorlesungsbegleitende Übung wird in zwei Formaten angeboten. Kolloquien werden in größeren Gruppen abgehalten, Tutorien in kleineren Gruppen. Kolloquien werden üblicherweise von Mitarbeiterinnen und Mitarbeitern des Lehrstuhls gehalten, in den Tutorien kommen Studierende höherer Semester zum Einsatz.

<b>Höhere Mathematik II</b>					
<b>Modul-Nr:</b> 150162 (Übung: 150163)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 2 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> V: 4 SWS; Ü: 2 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> Vorlesung: Unbegrenzt; Übung: 50 pro Gruppe
<b>Teilnahmevoraussetzungen:</b> Keine <b>Empfohlene Vorkenntnisse:</b>  Höhere Mathematik I					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls: Die Studierende <ul style="list-style-type: none"> <li>• Verstehen und wenden grundlegende mathematische Begriffe und Notationen an</li> <li>• Sind in der Lage die Themen zu erläutern und Einsatzmöglichkeiten zu benennen</li> <li>• Können Formalismen und Verfahren auswählen und ausführen, die erzielten Ergebnisse interpretieren</li> <li>• Übertragen und wenden zugehörige Lösungsverfahren auch auf praktische Probleme an</li> </ul> <b>Vermittelte Kompetenzen:</b> Kernkompetenzen: <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> </ul> Fachspezifische Kompetenzen: <ul style="list-style-type: none"> <li>• Mathematik</li> </ul>					
<b>Inhalt:</b> Potenzreihen und Fourierreihen, Vektorräume, Lineare Gleichungssysteme, Matrizen, Determinanten, Eigenwerte und Eigenvektoren, Differenzialrechnung (in $\mathbb{R}^n$ ), Gewöhnliche Differenzialgleichungen					
<b>Lehrformen:</b> Vortrag der Lehrenden in der Vorlesung (mit zum Teil digitalen Lehrformate), Gruppenarbeit in den Übungen, Ergänzung der Bearbeitung der Hausaufgaben in Einzel- oder Gruppenarbeit durch digitale Aufgaben, online Tests					

## KAPITEL 1. PFLICHTMODULE MATHEMATIK

<b>Prüfungsformen:</b> Klausur (120 Minuten)
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 9 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  PD Dr. Daniela Kacso

<b>Diskrete Mathematik I</b>					
<b>Modul-Nr:</b> 150308 (Übung: 150309)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 3 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung Diskrete Mathematik I  b) Übungen zu a)			<b>Kontaktzeit:</b> V: 4 SWS; Ü 2 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> Übung: 40 pro Gruppe
<b>Teilnahmevoraussetzungen:</b> Keine					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls					
<ul style="list-style-type: none"> <li>• haben die Studierenden abstrakte diskrete Strukturen kennengelernt und können diese sicher erläutern</li> <li>• können sie konkrete Problemstellungen mit den kennengelernten Strukturen modellieren</li> <li>• haben sie grundlegende algorithmische Techniken kennen- und anwenden gelernt und können diese Algorithmen analysieren</li> <li>• können sie logische Zusammenhänge zwischen den verschiedenen kennengelernten Konzepten erkennen und versteckte Anwendungsmöglichkeiten finden</li> </ul>					
<b>Inhalt:</b> Kombinatorik, Abzählprobleme; Grpahentheorie: Graphenexploration und weitere ausge-suchte Graphenprobleme; Grundkenntnisse in elementarer Zahlentheorie mit kryptographi-schen Anwendungen, Designtechniken für effiziente Algorithmen, Aufstellen und Lösen von Rekursionsgleichungen; grundlegende algebraische Strukturen mit Anwendungen auf symme-trische Zählprobleme und fehlerkorrigierende Codes					
<b>Lehrformen:</b> Vortrag der Lehrenden in der Vorlesung und Kleingruppenarbeit in den Übungen, Bear-beitung wöchentlicher Hausaufgaben einzeln oder in Kleingruppen					
<b>Prüfungsformen:</b> Klausur 180 Minuten					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					
<b>Stellenwert der Note für die Endnote:</b> 9 / 175					
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  PD Dr. Björn Schuster					

# Pflichtmodule Informatik

<b>Informatik 1 - Programmierung</b>					
<b>Modul-Nr:</b> 141328	<b>Credits:</b> 5 CP	<b>Workload:</b> 145 h	<b>Semester:</b> 1 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 100 h	<b>Gruppen- größe:</b> beliebig, zuletzt über 500 Teil- nehmer
<ul style="list-style-type: none"> <li>• Vorlesung (2 SWS)</li> <li>• Übung (1 SWS)</li> </ul>					
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Digitalisierung: Hausaufgaben werden automatisiert korrigiert und bepunktet. Es werden kurze Lernabschnitte im flipped-classroom Format eingefügt. Zur Aktivierung gibt es kurze Abstimmungen im Hörsaal. Die Vorlesung wird aufgezeichnet und als Video-Stream zur Verfügung gestellt.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen die Teilnehmer die wichtigsten Konzepte imperativer und objektorientierter Programmierung,</li> <li>• können die Teilnehmer eigene Programme schreiben,</li> <li>• verstehen die Teilnehmer erste Begriffe der Informatik wie etwa Laufzeit in Landau-Notation.</li> </ul>					
<p><b>Inhalt:</b> Imperative und objektorientierte Programmierung, sowie erste Grundbegriffe der Informatik</p>					
<p><b>Lehrformen:</b> Vorlesung und Übung</p>					
<p><b>Prüfungsformen:</b> Klausur 120 Minuten, Wiederholungsklausur nach 6 Monaten</p>					
<p><b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur</p>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Verwendung des Moduls (in anderen Studiengängen):</b> Bachelor Elektrotechnik Bachelor IT-Sicherheit
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Tobias Glasmachers</b> Prof. Dr. Tobias Glasmachers, Florian Fricke



## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Programmieren in C</b>					
<b>Modul-Nr:</b> 149872	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 45 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden beherrschen die grundlegenden Sprachkonstrukte von C mit Betonung der prozeduralen Betrachtungsweise und haben ein Verständnis für die Sicherheitsproblematik von C.</p> <p><b>Vermittelte Kompetenzen:</b> Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Software-Entwurfsmethoden</li> </ul> <p>Technologien:</p> <ul style="list-style-type: none"> <li>• C</li> <li>• Matlab</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Verfahren der Strukturierten Programmierung</li> <li>• Einführung in die Programmiersprache C (C90/C99/C11)</li> <li>• Elementare Sprachkonstrukte (Standard-Datentypen, Ausdrücke, Kontrollstrukturen)</li> <li>• Prozedurale Betrachtungsweise (Funktionen und Programmstrukturen)</li> <li>• Klassische Datenstrukturen (Arrays, Verbunde) und Zeiger</li> <li>• Dynamische Datenstrukturen</li> <li>• Sicherheitsproblematik</li> </ul>					
<b>Lehrformen:</b> Vorlesung (+Übung)					
<b>Prüfungsformen:</b> Klausur					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Stellenwert der Note für die Endnote: 3 / 175</b>
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Markus Dürmuth</b> Prof. Dr. Markus Dürmuth, André Feiler
<b>Sonstige Informationen:</b> <b>Literatur:</b> Skript

<b>Computernetze</b>					
<b>Modul-Nr:</b> 141250	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 2 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Computernetze			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> ca. 400 Studie- rende pro Vorle- sung/Übung
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Elementare Grundkenntnisse der Informatik</p>					
<p><b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende die wichtigsten Standards, die das heutige Internet verwendet.</li> <li>• kennen Studierende grundlegende Angriffskonzepte auf Computernetzwerke</li> <li>• verstehen Studierende den Zusammenhang zwischen den einzelnen Schichten eines Computernetzwerks und der darin enthaltenen Protokolle</li> <li>• können Studierende die wichtigsten Netzwerktools für Analysezwecke anwenden</li> </ul>					
<p><b>Inhalt:</b></p> <p>Die Vorlesung gibt eine Einführung in grundlegenden Protokolle und Anwendungen von Computernetzen. Der Schwerpunkt der Vorlesung liegt auf Standardprotokollen und -Algorithmen, wie sie in modernen Computernetzwerken (zum Beispiel im Internet) eingesetzt werden.</p> <p>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.</p> <p>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.</p> <p>Als Primärliteratur wird "Computernetzwerke: Der Top-Down Ansatz" von Kurose und Ross (Pearson Verlag) verwendet.</p>					
<p><b>Lehrformen:</b></p> <p>Moodle-Unterstützte Hausaufgaben mit praxisnahen, computerunterstützten Übungen. Tool-</p>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

der-Woche: Vorstellung, Einarbeitung, und Verwendung von Netzwerkrelevanten Computer-analysetools
<b>Prüfungsformen:</b> schriftliche Modulabschlussprüfung von 120 min
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Verwendung des Moduls (in anderen Studiengängen):</b> Studiengang Bachelor IT-Sicherheit (ET/IT), Studiengang Bachelor Angewandte Informatik
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  Christian Mainka
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. "Computernetzwerke", Kurose und Ross, Pearson.  2. "Computernetzwerke", Tanenbaum und Wetherall, Pearson.

<b>Informatik II</b>					
<b>Modul-Nr:</b> 141321	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Inhalte der Vorlesung Informatik I</p>					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden erhalten einen systematischen Überblick über Prinzipien, Methoden, Konzepte und Notationen von verschiedenen Algorithmen und Datenstrukturen. Dieses Wissen - verbunden mit den praktischen Übungen am Computer - befähigt die Studierenden, effiziente Programme problemgerecht zu entwickeln, zu analysieren, zu überprüfen und in die Programmiersprache C++ zu transformieren, zu übersetzen und bzw. darin auszuführen.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Graphentheorie</li> <li>• Programmieren</li> </ul> <p>Technologien:</p> <ul style="list-style-type: none"> <li>• C++</li> </ul>					
<p><b>Inhalt:</b> Die Vorlesung besteht aus vier größeren Blöcken, die wie folgt aufgeteilt sind: Im ersten Block werden Grundbegriffe eingeführt, der Fokus liegt auf dem Vergleich und der Bewertung von Algorithmen. Im zweiten Block werden klassische Sortialgorithmen (z: B. insertion sort, mergesort und quicksort) vorgestellt. Klassische Suchalgorithmen wie binäre Suche oder verschiedene Arten von Baumstrukturen werden im dritten Block vorgestellt. Im letzten Block wird ein Überblick über Graphalgorithmen und Operationen auf Strings gegeben. Begleitet wird die Vorlesung von Übungen, in denen die vorgestellten Konzepte und Techniken praktisch ausprobiert werden sollen.</p>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Lehrformen:</b> Vorlesung (+Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr.-Ing. Tim Güneysu</b>
<b>Sonstige Informationen:</b> Alle Materialien sind per Moodle verfügbar, bitte registrieren Sie sich dort für den Kurs.  <b>Literatur:</b> Die Veranstaltung basiert auf dem Buch "Algorithms" von Sedgewick und Wayne (siehe <a href="http://algs4.cs.princeton.edu/">http://algs4.cs.princeton.edu/</a> ).

<b>Objektorientierte Modellierung</b>					
<b>Modul-Nr:</b> 128047 (Übung: 128048)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Objektorientierte Modellierung			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Einführung in die Informatik und Programmierung					
<p><b>Lernziele (learning outcomes):</b>                      Globales Ziel dieser Veranstaltung ist es, einen systematischen Überblick über Prinzipien, Methoden, Konzepte und Notationen der Objektorientierte Modellierung zu geben. Dieses Wissen – verbunden mit den praktischen Übungen am Computersystem – soll den Studierenden befähigen, nach internationalen Standards konzipierte, effiziente Programme problemgerecht zu entwickeln, zu analysieren und zu überprüfen. Dazu gehört das Beschreiben von Sachverhalten mittels der UML (Unified Modeling Language) und das Arbeiten mit der Programmiersprache Java. Besonders mit fortschreitender Digitalisierung ist es zentral, Programmstrukturen entwickeln zu können, welche einfach zu warten sind und es ermöglichen, umfangreiche Teams an einem Softwareprojekt zu beschäftigen.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• Können Studierende in der Programmiersprache Java programmieren.</li> <li>• Können Studierende Objektorientierte Methoden auf Programmiersprachen abstrahieren.</li> <li>• Können Studierende geläufige Objektorientierte Muster erkennen, anwenden und programmieren.</li> <li>• Sind Studierende in der Lage, umfassende Programmstrukturen eigenständig zu entwerfen, sodass diese übersichtlich und wartbar sind.</li> <li>• Können Studierende grafische Benutzeroberflächen in Java entwerfen.</li> </ul>					
<p><b>Inhalt:</b>                      Basiskonzepte der Objektorientierung - Klassen und Objekte - Generalisierung - Vererbung und Schnittstellen - Assoziationen - Generische Datentypen und Container - Unified Modeling Language                      GUI-Programmierung - Ereignisverarbeitung - Model-View-Controller-Prinzip - Dialog- und E/A-Gestaltung</p>					
<b>Lehrformen:</b> Vorlesungen, digitale und vor Ort betreute Übungen					
<b>Prüfungsformen:</b>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

Schriftliche Klausur im Umfang von 120 Minuten
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Markus König</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. Balzert, Helmut, Priemer, Jürgen "Java 6: Anwendungen programmieren, 2. Auflage", W3l, 2010  2. Balzert, Helmut "Java:Objektorientiert programmieren, 2. Auflage", W3l, 2010



<b>Rechnerarchitektur</b>					
<b>Modul-Nr:</b> 141140	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Digitaltechnik, Programmiersprachen, Eingebettete Prozessoren</p>					
<p><b>Lernziele (learning outcomes):</b> Ziel dieser Lehrveranstaltung ist es, Zusammenhänge und Detailkenntnisse zum Aufbau, zu Komponenten und zur Funktionsweise moderner Computersysteme in Hard- und Software zu vermitteln. Damit soll für die Studierenden die Basis geschaffen werden, sowohl in der Computertechnik selbst, als auch in deren Anwendungsbereichen - wie z.B. der Automatisierungstechnik - Computerkomponenten und -systeme auslegen, und entwickeln zu können.</p> <p>Um die Studierenden zum Einen hinsichtlich Teamarbeit, Kommunikationsfähigkeit und Dokumentationsfähigkeit weiter zu qualifizieren und zum Zweiten anwendungsbezogene, praxisrelevante Problemstellungen und deren Lösungsmöglichkeiten zu vermitteln, wird veranstaltungsbegleitend ein Projekt angeboten, das im Team von 3 - 4 Studierenden zu bearbeiten ist. Abhängig von der inhaltlichen und formal-stilistischen Ausarbeitung kann ein Bonus von bis zu 10% erworben werden, der bei der Abschlussklausur angerechnet wird.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Kritikfähigkeit</li> <li>• Literaturrecherche und Dokumentation</li> <li>• Selbständiges Lernen und Arbeiten</li> <li>• Teamarbeit und Teamfähigkeit</li> <li>• Wissenschaftliches Arbeiten und Schreiben</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Logik und Schaltungen</li> </ul>					

### **Inhalt:**

Im ersten Teil der Veranstaltung werden, ausgehend von grundlegenden Computerstrukturen (Von-Neumann-Architektur, SISD, SIMD, MIMD), grundlegende Fähigkeiten zum anforderungsgerechten Entwurf, und zur anwendungsbezogenen Realisierung von Computersystemen vermittelt. Konkrete Beispiele heutiger Computer für unterschiedliche Anwendungsfelder (8051, Pentium 4, Core-Architektur, Ultra Sparc III) runden die generellen Wissensinhalte ab. Einen besonderen inhaltlichen Schwerpunkt bildet die Programmierung der Mikroarchitekturebene als Ergänzung zu anderen Lehrveranstaltungen im Bereich der Informatik / Computertechnik (Programmiersprachen, Eingebettete Prozessoren). Im zweiten Teil der Veranstaltung werden die Basisfunktionen moderner Betriebssysteme behandelt. Schwerpunkte sind hier die Organisation von Prozessen mit Prozessscheduling und Interprozesskommunikation sowie die Behandlung von Deadlocks.

Im Detail ist die Lehrveranstaltung wie folgt gegliedert:

- Einführung
- Grundstrukturen und Definitionen
- Prinzipien moderner Computerarchitektur
- Struktur und Aufbau von Computersystemen
- Klassische "Von-Neumann-Struktur"
- Parallelitätsprinzipien
- Klassifikation und Merkmale von Computerarchitekturen
- Logisch digitale Ebene
- CPU-Chips und Busse
- Schnittstellen
- Mikroarchitekturebene
- Fallbeispiel einer Mikroarchitektur
- Design der Mikroarchitekturebene
- Methoden der Leistungsoptimierung
- Beispiele der Mikroarchitekturebene
- Betriebssystemebene
- Prozesse und Threads (Scheduling, Interprozesskommunikation)
- Deadlock-Behandlung
- Organisation virtueller Speicher
- Virtuelle E/A-Instruktionen

## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Lehrformen:</b> Vorlesung (Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Studiendekan ETIT</b> Herr M. Sc. Florian Fricke
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. Tanenbaum, Andrew S. "Computerarchitektur. Strukturen - Konzepte - Grundlagen", Pearson, 2006  2. Tanenbaum, Andrew S. "Modern Operating Systems", Pearson, 2009  3. Tanenbaum, Andrew S. "Moderne Betriebssysteme", Pearson, 2009  4. Patterson, David A., Hennessy, John L., Bode, Arndt "Rechnerorganisation und -entwurf", Spektrum Akademischer Verlag, 2005  5. Tanenbaum, Andrew S. "Structured Computer Organization", Prentice Hall, 2005

<b>Theoretische Informatik</b>					
<b>Modul-Nr:</b> 150240 (Übung: 150241)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 3 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 6 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> Ca. 120 Stu- dierende in der Vorlesung, Ca 30 Studierende in der Übung
<b>Teilnahmevoraussetzungen:</b> Elementare Grundkenntnisse in der Informatik.					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> <li>• kennen Studierende grundlegende Arten von Automaten und formalen Sprachen</li> <li>• können Studierende formale Sprachen in die Chomsky-Hierarchie einordnen und einen erkennenden Automaten dafür angeben</li> <li>• verstehen Studierende den Begriff der Berechenbarkeit</li> <li>• können Studierende für einfache Funktionen Turingmaschinen angeben, die diese be- rechnen</li> <li>• verstehen Studierende grundlegenden Konzepte der algorithmischen Komplexität von Problemen, insbesondere die Komplexitätsklassen P und NP, sowie den Begriff der NP- Schwerheit</li> </ul>					
<b>Inhalt:</b> Die Vorlesung liefert eine Einführung in die Theorie der Grammatiken (insbesondere die Chomsky-Hierarchie) und Automaten (endlicher Automat, Kellerautomat, Turing-Maschine). Sie gibt ferner einen Einblick in die Berechenbarkeits- und NP-Vollständigkeitstheorie. Hierbei geht es um die Frage, welche Rechenprobleme (überhaupt bzw. mit vertretbarem Aufwand) gelöst werden können. Dabei zeigt sich, dass es inhärent schwere Probleme gibt, die von Rechnern nicht zufriedenstellend gelöst werden können.					
<b>Lehrformen:</b> Vorlesung (als Tafelvortrag) und Übungen					
<b>Prüfungsformen:</b> Klausur über 180 Minuten					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					

<b>Verwendung des Moduls (in anderen Studiengängen):</b> Angewandte Informatik, Informatik, Mathematik, IT-Sicherheit
<b>Stellenwert der Note für die Endnote:</b> 9 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Maike Buchin</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b> Die Vorlesung orientiert sich hauptsächlich an dem Buch "Theoretische Informatik - kurzgefasst" von Uwe Schöning (HTb, 2001). Zur Vertiefung des Stoffes kann sowohl das Buch "Introduction to Automata Theory, Languages, and Computation" von John E. Hopcroft, Rajeev Motwani und Jeffrey D. Ullman (Addison Wesley, 2000) als auch "Introduction to the Theory of Computation" von M. Sipser (PWS Publishing Company, 1997) dienen.

<b>Datenstrukturen</b>					
<b>Modul-Nr:</b> 150322 (Übung: 150323)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 6 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> Ca. 120 Stu- dierende in der Vorlesung, ca. 30 Studierende in der Übung
<b>Teilnahmevoraussetzungen:</b> Die Kenntnis einer höheren Programmiersprache ist hilfreich, aber nicht im engen Sinne erforderlich.					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> <li>• kennen Studierende grundlegende Datenstrukturen und Algorithmenparadigmen</li> <li>• können Studierende die Korrektheit eines Algorithmus formal beweisen</li> <li>• können Studierende die Laufzeit und den Speicherbedarf von Algorithmen und Datenstrukturen analysieren und bewerten</li> <li>• können Studierende Algorithmen und Datenstrukturen für spezifische Probleme entwickeln</li> </ul>					
<b>Inhalt:</b> Die Vorlesung gibt einen systematischen Überblick über den Entwurf und die Analyse von Algorithmen und Datenstrukturen. Dazu werden zunächst grundlegenden Methoden der Analyse (insbesondere Korrektheit, Laufzeit und Speicherbedarf) von Algorithmen vorgestellt. Anschließend sehen wir einige Algorithmen zum Sortieren und Suchen. Ebenfalls werden verschiedene grundlegende Datenstrukturen (Listen, Felder, Suchbäume und Heaps) vorgestellt. Schließlich betrachten wir Graphen: ihre Darstellung und diverse Algorithmen auf Graphen (Durchläufe, Kürzeste Wege, Minimale Spannbäume). In den Übungen lernen die Studierende sowohl die theoretische Analyse von Algorithmen und Datenstrukturen als auch deren praktische Umsetzung in einer modernen Programmiersprache (z.B. Python oder C++).					
<b>Lehrformen:</b> Vorlesung (als Folien- oder Tafelvortrag) und Übung					
<b>Prüfungsformen:</b> Klausur über 180 Minuten					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					

<b>Verwendung des Moduls (in anderen Studiengängen):</b> Angewandte Informatik, Informatik, Mathematik, IT-Sicherheit
<b>Stellenwert der Note für die Endnote:</b> 9 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Maike Buchin</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. Güting, Ralf Hartmut und Dieker, Stefan. Datenstrukturen und Algorithmen. Teubner.  2. Aho, Hopcroft und Ullman. The design and analysis of computer algorithms. Addison-Wesley.

<b>Software-Engineering</b>					
<b>Modul-Nr:</b> 260085	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übungen			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 105 h	<b>Gruppen- größe:</b> 15 pro Übungs- gruppe. 3-4 Studierende pro Fallbeispielauf- gabe
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Objektorientierte Modellierung, Programmierkenntnisse.</p>					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden</p> <ul style="list-style-type: none"> <li>• können Grundlagen, Aktivitäten und Methoden des Requirements-Engineering und von Softwareentwicklungsprozessen erläutern und einordnen</li> <li>• verfügen über vertiefte Kenntnisse über ausgewählte Aspekte des Softwareentwicklungsprozesses</li> <li>• verfügen über Grundkenntnisse zum Thema Softwarequalität</li> <li>• kennen und verstehen die grundsätzlichen Ziele und Verantwortlichkeiten im Software-Life-Cycle</li> <li>• kennen und verstehen die verschiedenen Aktivitäten innerhalb des Software-Life-Cycles und deren Abhängigkeiten</li> </ul> <p><b>Vermittelte Kompetenzen:</b> Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Entwicklungsprozesse</li> <li>• Software-Entwurfsmethoden</li> </ul> <p>Technologien:</p> <ul style="list-style-type: none"> <li>• UML</li> <li>• Eclipse</li> </ul>					
<b>Inhalt:</b>					



## KAPITEL 2. PFLICHTMODULE INFORMATIK

Die Studierenden lernen unterschiedliche Formen von (klassischen und agilen) Vorgehensmodellen in der Softwareentwicklung kennen. Sie lernen Methoden der Anforderungserhebung, des Entwurfs und des Testens kennen und setzen diese in einem realen Fallbeispiel selbstständig um.

Die Vorlesung wird von Experten aus der Praxis als Gastdozenten begleitet vertiefende Einblicke in Aufgabenfelder und aktuelle Forschungsthemen zu erhalten.

### **Lehrformen:**

Die Inhalte werden in Vorlesungen vermittelt und in kleinen Übungsgruppen vertieft und eingeübt. Bei letzterem werden alle Methoden und Kenntnisse für die Bearbeitung des Fallbeispiels behandelt. Die Studierenden präsentieren zweimal im Semester ihre Ergebnisse beim Fallbeispiel.

### **Prüfungsformen:**

Schriftliche Klausur 60 min. / Zwei Fallbeispielberichte (jeweils ca. 20 Seiten und Bearbeitungszeit ca. 4 Wochen) in Gruppenarbeit / Präsentation der Zwischenergebnisse (jeweils ca. 10 Minuten Vortrag).

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Endnote setzt sich zusammen aus den Fallbeispiel-Abgabe (jeweils 25% der Endnote) und der schriftliche Klausur (50% der Endnote)

**Stellenwert der Note für die Endnote:** 5 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Thomas Herrmann**  
Marcel Schmittchen Msc.

### **Sonstige Informationen:**

#### **Literatur:**

1. Ian Sommerville: Software Engineering; Addison-Wesley, 2001 (6th edition)
2. Klaus Pohl: Requirements Engineering: Grundlagen, Prinzipien, Techniken, dpunkt.verlag, 2. Aufl., 2008

<b>Web-Engineering</b>					
<b>Modul-Nr:</b> 128968	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b>                      Die Entwicklung von Web-Anwendungen und -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 Softwareprojekten unerlässlich.                      Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende gängige Konzepte der Web-Entwicklung in den Aspekten Präsentation, Transport und Bereitstellung von Daten;</li> <li>• beherrschen Studierende grundlegende Fähigkeiten in Webseitendarstellung, dynamischen Web-Anwendungen und modernen Services (Node.js).</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Einführung in clientseitige Web-Entwicklung: HTML, CSS, JavaScript, Web Components;</li> <li>• Transportverfahren und deren Nutzung: Representational State Transfer (REST), Asynchronous JavaScript and XML (AJAX);</li> <li>• Serverseitige Entwicklung mit Node.js und weiterführende Technologien.</li> </ul>					
<p><b>Lehrformen:</b> Vorlesung und Übung mit Gruppenarbeiten am Rechner.</p>					
<p><b>Prüfungsformen:</b> Schriftliche Prüfung von 120 min.</p>					
<p><b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur</p>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

<b>Verwendung des Moduls (in anderen Studiengängen):</b> Masterstudiengang ITS (Wahlfach)
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr.-Ing. Markus König</b> Henk Freimuth
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. jQuery - Das umfassende Handbuch, Rheinwerk Verlag, 2013, ISBN 978-3-8362-2638-7 2. Node.js - Das umfassende Handbuch, Rheinwerk Verlag, 2013, ISBN 978-3-8362-2119-1

<b>Betriebssysteme</b>					
<b>Modul-Nr:</b> 141246	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Grundkenntnisse der Informatik</p> <p><b>Empfohlene Vorkenntnisse:</b>  Grundkenntnisse der Informatik</p>					
<p><b>Lernziele (learning outcomes):</b> Ziel dieser Lehrveranstaltung ist es, ein solides Grundverständnis von modernen Betriebssystemen, ihrer Funktion und ihrer Implementierung zu vermitteln.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Selbständiges Lernen und Arbeiten</li> <li>• Teamarbeit und Teamfähigkeit</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmen</li> <li>• IT-Sicherheit</li> </ul> <p>Technologien:</p> <ul style="list-style-type: none"> <li>• C</li> </ul>					
<p><b>Inhalt:</b> Es werden die wichtigsten Grundlagen zu Betriebssystemen vorgestellt. Dazu gehören zum Beispiel:</p> <ul style="list-style-type: none"> <li>• Betriebssystemkonzepte</li> <li>• Prozesse und Threads, Interprozesskommunikation</li> <li>• Scheduling-Mechanismen</li> </ul>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

- Speicherverwaltung, Speicherabstraktionen, Paging
- Dateisysteme
- Eingabe- und Ausgabeverwaltung
- Algorithmen zur Vermeidung von Deadlocks
- Grundlagen der Sicherheit von Betriebssystemen

Gegen Ende der Vorlesung werden (wenn die Zeit es erlaubt) speziellere Themen behandelt, wie zum Beispiel Multimedia-Betriebssysteme, Multiprozessorsysteme, und Entwurf von Betriebssystemen.

Ergänzend zur Vorlesung werden Übungsaufgaben gestellt und in der Übungsstunde besprochen.

Um den Bezug zu modernen Betriebssystemen (aktuellen Versionen von Linux, Windows, und Mac OS) herzustellen, werden alle Themen an praktischen Beispielen illustriert. Dies ermöglicht es den Studierenden, die in der Vorlesung besprochenen Themen praktisch nachzuvollziehen.

**Lehrformen:**

Vorlesung (+Übung)

**Prüfungsformen:**

Klausur

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote:** 5 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. rer. nat. Thorsten Holz**

**Sonstige Informationen:**

Wird das erste Mal im Sommersemester 2015 angeboten.

**Literatur:**

1. Stallings, W.: Operating Systems: Internals and Design Principles, Pearson Verlag
2. "Moderne Betriebssysteme", Andrew S. Tanenbaum, Pearson 2012.
3. "Modern Operating Systems", Andrew S. Tanenbaum, Pearson 2012.

<b>Datenschutz</b>					
<b>Modul-Nr:</b> 260081	<b>Credits:</b> 4 CP	<b>Workload:</b> 120 h	<b>Semester:</b> 5 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung mit Folien Übungen: Wissens- und Verständnis- abfragen, Anwendung auf Beispiele			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> Max 25 Übun- gen Vorlesung 120 Studierende
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> 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.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende die Grundzüge des Datenschutzrechtes,</li> <li>• verstehen Studierende die gesellschaftlichen Hintergründe,</li> <li>• können Datenverarbeitungsprozesse hinsichtlich der Relevanz des Datenschutzrechtes analysieren und</li> <li>• können Lösungsmuster anwenden um Systeme datenschutzfreundlich und datenschutzrechtskonform zu gestalten.</li> </ul> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Selbständiges Lernen und Arbeiten</li> </ul>					

## KAPITEL 2. PFLICHTMODULE INFORMATIK

- Teamarbeit und Teamfähigkeit
- Präsentation wissenschaftlicher Ergebnisse

Fachspezifische Kompetenzen:

- Rechtliche Kompetenzen

### **Inhalt:**

- 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: Wissens- und Verständnisabfragen, Anwendung auf Beispiele

### **Prüfungsformen:**

Klausur 90min Bonuspunkte für erfolgreiche Übungsteilnahme

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

### **Verwendung des Moduls (in anderen Studiengängen):**

Wahlmodul in Studiengängen der IT-Sicherheit

**Stellenwert der Note für die Endnote:** 4 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Dr. Kai-Uwe Loser**  
Dr. Martin Degeling

### **Sonstige Informationen:**

#### **Literatur:**

1. Gola, Peter, Jaspers, Andreas "Das BDSG im Überblick", Datakontext Fachverlag G, 2006

## KAPITEL 2. PFLICHTMODULE INFORMATIK

2. Tinnefeld, Marie-Theres, Ehmann, Eugen, Gerling, Rainer W. "Einführung in das Datenschutzrecht. Datenschutz und Informationsfreiheit in europäischer Sicht", Oldenbourg, 2004



<b>Datenbanksysteme</b>					
<b>Modul-Nr:</b> 150304 (Übung: 150305)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 5 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Datenbanksysteme (Vorlesung)  b) Datenbanksysteme (Übung)			<b>Kontaktzeit:</b> 6 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine <b>Empfohlene Vorkenntnisse:</b>  Grundlagen der Informatik und Datenstrukturen.					
<b>Lernziele (learning outcomes):</b> Die Studierenden lernen den in (4+2)-stündigen Lehrveranstaltungen über Datenbanksysteme üblichen Stoff. Die Entwurfstheorie nimmt dabei einen großen Raum ein. Die Benutzung von Datenbanken aus Anwendersicht ist kein Lehrschwerpunkt. Die Studierenden sollen in die Lage versetzt werden, künftig selbst effiziente Datenbanken zu konzipieren, die Anwender nutzen können.					
<b>Inhalt:</b>  <ul style="list-style-type: none"> <li>• Strukturen der physikalischer Ebene</li> <li>• Entity-Relationship-Diagramme</li> <li>• Sprachen und Kalküle zur Datendefinition und -manipulation</li> <li>• Optimierung von Abfragen</li> <li>• Deduktionssysteme bei Datenbanken</li> <li>• Strategien der Parallelisierung und der Fehlerbehandlung</li> <li>• Entwurfstheorie: Normalformen, funktionale Anhängigkeiten, verlustlose und abhängigkeitserhaltene Zerlegungen</li> </ul>					
<b>Lehrformen:</b> Vortrag in der Vorlesung, Aufgabenzettel zur Bearbeitung, Aufgabenbesprechung in den Übungen					
<b>Prüfungsformen:</b> 90 minütige Klausur am Ende der Vorlesungszeit, Nachschreibeklausur zu Anfang des Folgesemesters					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					

<b>Stellenwert der Note für die Endnote: 9 / 175</b>
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Dr. Edgar Korthauer</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. A. Kemper, A. Eickler "Datenbanksysteme", Oldenbourg Verlag, 2009  2. R. Elmasri, S. Navathe "Grundlagen von Datenbanksystemen", Pearson, 2009

# Pflichtmodule Betriebswirtschaftlehre

<b>Wirtschaftlichkeitsanalyse</b>					
<b>Modul-Nr:</b> 076000 (Übung: 076001)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 1 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung mit integrierter Übung			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 110 h	<b>Gruppen- größe:</b> ca. 150
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls:</p> <ul style="list-style-type: none"> <li>• sollen die Studierenden Grundbegriffe der Wirtschaftlichkeitsanalyse kennen</li> <li>• sollen die Studierenden die verschiedenen Teilgebiete der Wirtschaftlichkeitsanalyse auseinanderhalten können</li> <li>• sollen die Studierenden Aufgaben der Kostenarten-, Kostenstellen- und Kostenträgerrechnung lösen können</li> <li>• sollen die Studierenden Erfolgsgrößen errechnen und beurteilen können</li> <li>• sollen die Studierenden Methoden der Investitionsrechnung sicher anwenden</li> <li>• sollen die Studierenden kompetent mit wirtschaftlichen Fachbegriffen umgehen können</li> <li>• sollen die Studierenden in der Lage sein, sich mit Kaufleuten inhaltlich über die entsprechenden Themen auszutauschen</li> <li>• sollen die Studierenden bei Budgetverantwortung und eigenen Projekten die Grundbegriffe der Wirtschaftlichkeit kennen und berücksichtigen können</li> </ul> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> </ul>					

## KAPITEL 3. PFLICHTMODULE BETRIEBSWIRTSCHAFTLEHRE

- Fächerübergreifendes Denken und Arbeiten
- Kritikfähigkeit

Fachspezifische Kompetenzen:

- Rechnungswesen
- Wirtschaftliches Handeln

### **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:**

Schriftliche Modulabschlussprüfung von 60 Minuten

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote:** 5 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Dr. Barbara Wischermann**

<b>Einführung Management Science</b>					
<b>Modul-Nr:</b> 072120 (Übung: 072121/072122/072123)	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Der vorherige Erwerb betriebswirtschaftlicher Grundkenntnisse und der parallele Besuch des Moduls Projektmanagement werden empfohlen.</p>					
<p><b>Lernziele (learning outcomes):</b> Dieses Modul bietet eine Einführung in Management Science mit besonderem Schwerpunkt auf mathematischen Optimierungsmethoden. Studierende vertiefen ihre mathematischen und analytischen Fähigkeiten und sind in der Lage, Probleme zu strukturieren und ggf. zu modellieren. Für wichtige Problemstrukturen werden Methoden zur Lösung vermittelt. Studierende erwerben auf Basis der fachinhaltlichen Kenntnisse die Fähigkeit, neue Aufgaben in begrenzter Zeit selbstständig zu lösen.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Selbständiges Lernen und Arbeiten</li> <li>• Projekt- und Zeitmanagement</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Optimierungsmethoden</li> <li>• Mathematik</li> <li>• Wirtschaftliches Handeln</li> </ul>					
<p><b>Inhalt:</b></p> <p>In Einführung Management Science werden insbesondere Methoden und Modelle der Unternehmensforschung intensiv behandelt. Diese werden in der Praxis zur Planung und zur Problemlösung in unterschiedlichen Anwendungsbereichen angewandt. Dazu zählen optimierende Methoden z.B. zur Produktionsprogrammplanung in der Fertigung, zur Investitions- und Finanzierungsplanung oder zur Lösung von Transport- bzw. Zuordnungsproblemen in Vertrieb, Logistik und Personalplanung. Die Unternehmensplanung bedient sich ebenfalls häufig quantitativer Methoden. Zusätzlich zur Vorlesung erfolgt in der begleitenden Übung die Vertiefung</p>					

### KAPITEL 3. PFLICHTMODULE BETRIEBSWIRTSCHAFTLEHRE

durch Anwendung der Methoden. Die Vorstellung von Standardsoftware zur Lösung größerer Problemstellungen und deren Einsatz ist ebenfalls Bestandteil der Veranstaltung.

Die Modulnote entspricht der Note der Klausur, welche aus Inhalten der Veranstaltungen gebildet wird.

**Lehrformen:**

Vorlesung (+Übung)

**Prüfungsformen:**

Klausur

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote:** 6 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Brigitte Werners**

**Sonstige Informationen:**

Das Modul wird über die Fakultät für Wirtschaftswissenschaft angeboten und dort als Teilmodul des Moduls "Quantitative Decision Making" geführt.

**Literatur:**

Veranstaltungsunterlagen und Literaturhinweise werden über Blackboard bereitgestellt.

# Nichttechnische Wahlmodule

<b>Nichttechnische Wahlmodule</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> immer	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> frei wählbar; Liste mit empfohlenen Veranstaltungen unter <a href="http://www.ai.rub.de">www.ai.rub.de</a>			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 105 h	<b>Gruppengröße:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Abhängig von der gewählten Veranstaltung.</p>					
<p><b>Lernziele (learning outcomes):</b> Die Teilnehmer erwerben so genannte Schlüsselfähigkeiten in den nichttechnischen Wahlfächern.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul>					
<p><b>Inhalt:</b> Die nichttechnischen Wahlfächer erweitern die Soft Skills. 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.</p>					
<p><b>Lehrformen:</b> frei wählbar; Liste mit empfohlenen Veranstaltungen unter <a href="http://www.ai.rub.de">www.ai.rub.de</a></p>					
<p><b>Prüfungsformen:</b> Abhängig von der gewählten Veranstaltung.</p>					
<p><b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur</p>					

<b>Stellenwert der Note für die Endnote: 5 / 175</b>
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Laurenz Wiskott</b> Dozenten der RUB
<b>Sonstige Informationen:</b> Unter anderem bieten sich die folgenden Fächer an: <ul style="list-style-type: none"><li>• Englisch für Studierende der Angewandten Informatik</li><li>• Andere nichttechnische Veranstaltungen</li></ul> Aus der Fakultät Wirtschaftswissenschaften dürfen nur die Lehrveranstaltungen belegt werden, die in der Liste (unter <a href="http://www.ai.rub.de">www.ai.rub.de</a> ) aufgeführt sind. Diese müssen über FlexNow innerhalb der Anmeldephase angemeldet werden.



# Praktische Fächer

<b>Intensivkurs C++</b>					
<b>Modul-Nr:</b> 310032	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 3 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Intensivkurs C++			<b>Kontaktzeit:</b> 2-wöchiger Blockkurs Keine SWS	<b>Selbststudium:</b> (ganztägige Präsenzver- anstaltung) h	<b>Gruppen- größe:</b> 20 - 40 Studie- rende
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Ziel dieses Kurses ist es, die wesentlichen Paradigmen der Programmiersprache C++ zu erarbeiten und am praktischen Beispiel selbst anzuwenden. Da die Teilnehmer auf einem sehr unterschiedlichen Vorkenntnisstand beginnen, ist es Teil der Praxisaufgaben, noch unbekannte aber für die Aufgaben notwendige Inhalte selbst oder mit Hilfe eines Betreuers zu erschließen. Erfolgreiche Teilnehmer dieses Kurses werden sich in der Lage sehen, kleine bis mittelgroße Softwareprojekte in C++ selbst zu planen, zu konzipieren und zu implementieren, wobei Designentscheidungen getroffen werden, die mit Wiederverwendbarkeit, Lesbarkeit, Effizienz oder Sicherheit begründet werden können.</p>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• die imperative C++-Programmiersprache</li> <li>• Zeigerarithmetik</li> <li>• Klassen</li> <li>• Vererbung</li> <li>• Templates</li> <li>• Die Standard Template Library und Input-Output-Streams</li> <li>• Features des C++14/C++17-Standards</li> </ul>					

## KAPITEL 5. PRAKTISCHE FÄCHER

**Lehrformen:**

Der Blockkurs beginnt täglich mit einer Vorlesung, die in die Thematik des Tages einführt, gefolgt von praktischen Programmierübungen für den Rest des Tages. Diese Aufgaben können allein oder in kleinen Gruppen bearbeitet werden. Zur Kontrolle sollen die Teilnehmer das von ihnen entwickelte Programm vorstellen und Grenzfälle aufzeigen. Zwei Tage des Kurses sind zum Aufholen nicht fertiggestellter Programmieraufgaben vorgesehen. An diesen Tagen werden keine neuen Themen eingeführt.

**Prüfungsformen:**

Die entwickelten Programme für jeden Tag werden benotet. Der Durchschnitt der Benotungen für jeden Tag ergibt die Gesamtnote für die Prüfung.

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Neben der erfolgreichen Bearbeitung der Einzeltagesaufgaben als Prüfungsleistung gibt es keine weiteren Voraussetzungen.

**Stellenwert der Note für die Endnote:** 3 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

Jun.-Prof. Sebastian Houben, Daniela Horn, M.A., M.Sc.

<b>Introduction to Python</b>					
<b>Modul-Nr:</b> 310533	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 2 weeks (block)
<b>Lehrveranstaltungen:</b> Programmierpraktikum: „Introduction to Python“			<b>Kontaktzeit:</b> 3.2 SWS	<b>Selbststudium:</b> 42 h	<b>Gruppen- größe:</b> 10-36
<p><b>Teilnahmevoraussetzungen:</b> Vorlesung "Informatik 1" und "Programmieren in C"</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>We expect fluency in one other programming language and familiarity with concepts like - loops and control structures (while, for, if ...), - basic data types and structures (boolean, int, float, string, arrays, ...), - functions, and - object-oriented programming.</p> <p>These concepts will not be taught separately. A solid understanding of basic maths and algorithms is also recommended for a successful project.</p>					
<p><b>Lernziele (learning outcomes):</b> After the successful completion of this course the students</p> <ul style="list-style-type: none"> <li>• will know and be able to apply basic syntax and structure of Python 3,</li> <li>• will understand numerical representations and processing of data using Numpy,</li> <li>• will have gained first practical experience in planning and conducting a small project in a team using Python 3.</li> </ul>					
<p><b>Inhalt:</b></p> <p>Python is a programming language that is wide-spread among scientists due to its readability and powerful standard libraries. This practical course teaches Python 3 to students with prior experience in other programming languages. In addition to introducing the language itself, we will focus on scientific computing including vectors and matrices as well as data processing and possibly simple machine learning. All course-work is done in teams of two.</p> <p>During the first week, participants will work on Jupyter notebooks autonomously and discover Python 3 in a largely self-taught manner. Teaching assistants are present and support is provided if required.</p> <p>During the second week, participants will implement a project in Python 3 using the previously acquired skills. We provide a default project, usually from the area of machine learning. Alternatively, own project ideas can be realized if discussed early on with the Teaching Assistants.</p>					
<p><b>Lehrformen:</b> Two week lab-course (Blockkurs in der vorlesungsfreien Zeit)</p>					
<p><b>Prüfungsformen:</b> Team-project and -presentation with additional individual questions. Requirement for project/presentation is successful completion of the notebook exercises.</p>					

## KAPITEL 5. PRAKTISCHE FÄCHER

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Successfully demonstrating command of Python/numpy in project presentation and questions.

**Verwendung des Moduls (in anderen Studiengängen):**

The lab course is open for students of other faculties, if they fulfill the requirements. They will be graded similar to students from Applied Computer Science. If there are more applicants than seats, Applied Computer Science students will be treated with priority. After that, the seats will be allocated on a first-come, first-serve basis based on enrollment emails (see below).

**Stellenwert der Note für die Endnote:** 3 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

Merlin Schüler, Prof. Dr. rer. nat. Laurenz Wiskott

**Sonstige Informationen:**

The course language is English.

In addition to registering via your examination office/FlexNow/..., please send an eMail to [python@ini.rub.de](mailto:python@ini.rub.de) Include your name, your student ID, and your study program.

<b>Programmierpraktikum Android</b>					
<b>Modul-Nr:</b> 260086	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 3-4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Block-Praktikum			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> 14-20
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b>                      Das Praktikum richtet sich an Studierende mit Vorkenntnissen in Java (oder einer vergleichbaren Objekt-orientierten Sprache) und soll den Einstieg in die App-Entwicklung für Android ermöglichen. Für die Entwicklung sollen die Teilnehmer auf eigene Hardware (PC und bestenfalls Android Smartphone) zurückgreifen. Die Studierenden lernen nach kurzer Einführung in das Themengebiet „Android-App-Entwicklung“ selbstständig Lösungen für häufige Probleme zu entwickeln. Dabei wird das Nutzen komplexer Dokumentation erlernt, sowie die Recherche bei spezifischen Problemen sowie die Nutzung von Entwicklerwerkzeugen. Neben den spezifischen Inhalten, werden Themen wie Fehlerbehandlung, Internationalisierung, Modularisierung, Usability/User-Expierence und Barrierefreiheit vertieft.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Literaturrecherche und Dokumentation</li> <li>• Projekt- und Zeitmanagement</li> <li>• Selbständiges Lernen und Arbeiten</li> <li>• Teamarbeit und Teamfähigkeit</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Software-Entwurfsmethoden</li> <li>• Programmieren</li> </ul> <p>Technologien:</p> <ul style="list-style-type: none"> <li>• Python</li> <li>• Android Studio</li> </ul>					
<p><b>Inhalt:</b>                      Auszug Themen: „Android Studio“ IDE, Views, Activities, Intents, Gestures, RESTful Webservices, NFC.</p>					
<p><b>Lehrformen:</b>                      Das Praktikum wird als einwöchige Blockveranstaltung in der vorlesungsfreien Zeit durch-</p>					

## KAPITEL 5. PRAKTISCHE FÄCHER

geführt. Nach kurzen Einführungsvorträgen durch die DozentInnen arbeiten die Studierenden selbstständig zusammenhängende Praktikumsaufgaben ab. Jede TeilnehmerIn erarbeitet eine eigenständige Lösung, der Austausch mit anderen TeilnehmerInnen und gegenseitige Hilfe sind dabei gewünscht. Nach jeder Teilaufgabe erhalten die TeilnehmerInnen Feedback zu ihrer Lösung von den DozentInnen.

**Prüfungsformen:**

Die Praktikumsaufgaben werden durch die DozentInnen bewertet.

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Regelmäßige Teilnahme am Praktikum und Bearbeitung der Aufgaben.

**Stellenwert der Note für die Endnote:** 3 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

Christopher Lentzsch M.Sc.

<b>Grundlagen Roboterprogrammierung</b>					
<b>Modul-Nr:</b> 138970	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Woche
<b>Lehrveranstaltungen:</b>  a) Praktikum			<b>Kontaktzeit:</b> 24h keine SWS	<b>Selbststudium:</b> 66 h	<b>Gruppen- größe:</b> 6 Studierende
<p><b>Teilnahmevoraussetzungen:</b> Keine  <b>Empfohlene Vorkenntnisse:</b></p> <p>Teilnahme an der Vorlesung "Grundlagen der Automatisierungstechnik"</p>					
<p><b>Lernziele (learning outcomes):</b>  Das Praktikum "Grundlagen der Roboterprogrammierung" soll den AI-Studierenden die Möglichkeit bieten, in einem komprimierten Kurs die Roboterprogrammierung an Industrierobotern zu erlernen. Ziel des einwöchigen Kurses ist die selbständige Entwicklung und Umsetzung von einfachen Roboterbewegungsprogrammen mit Peripherie-Interaktion, bspw. zur Greifer- und Sensorik-Integration.</p> <p>Mit den erlernten Fähigkeiten werden die Studierenden in die Lage versetzt, Industrieroboter zu programmieren sowie industrielle Aufgabenstellungen in der robotergestützten Automatisierung an einem realen System umzusetzen und es wird ein geeignetes Wissen für die mögliche spätere wissenschaftliche Vertiefung sowie die Berufstätigkeit im Automatisierungs- und Produktionssystemsbereich erlangt. Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• sind die Studierenden in der Lage Industrieroboter so zu programmieren, dass einfache Peripherie-Interaktionen möglich sind</li> <li>• können die Studierenden selbstständig Aufgabenstellungen entwickeln, die durch geeignete Roboterprogrammierung gelöst werden können</li> <li>• können die Studierenden Ihre abgeschlossenen Projekte in einer Präsentation unter Verwendung des entsprechenden Fachvokabulars vorstellen</li> </ul>					
<p><b>Inhalt:</b></p> <p>Diese Inhalte werden den Studierenden schrittweise und angeleitet näher gebracht und in selbstständiger Arbeit vertieft: Anfangen von der Arbeitssicherheit beim Umgang mit Industrierobotern sowie dem Aufbau und der Funktionsweise eines Robotersystems, über das Erstellen von einfachen Bewegungsprogrammen und dem Einmessen von Werkzeugen und Koordinatensystemen, endet der Kurs mit der Realisierung einer Handhabungsaufgabe, in der die Studierenden ein vollständiges Roboterprogramm inklusive Greiferansteuerung am Robotersystem umsetzen. Zu Beginn des Kurses werden die einfacheren Programme direkt per Teach-In am Roboter programmiert und ausgetestet. Mit der Steigerung der Komplexität der Roboterprogramme erfolgt die Programmierung dann in der Robotersprache RAPID, jeweils mit abschließenden Überprüfungen am realen System.</p> <p>Im Bereich der Kompetenzen im Kontext Digitalisierung ist der Simulationsteil angesiedelt, dieser beinhaltet die Einführung in ein kommerziell verfügbares und industriell einge-</p>					

## KAPITEL 5. PRAKTISCHE FÄCHER

setztes 3D-Robotersimulations- und Offline-Programmiersystem.
<b>Lehrformen:</b> Praktikum in 6er-Gruppe
<b>Prüfungsformen:</b> Präsentation der programmierten Aufgaben
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Erfolgreiche Beteiligung an der Gruppenarbeit und der -abschlusspräsentation
<b>Stellenwert der Note für die Endnote:</b> 1,71%
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  Dr. Alfred Hypki



**Verantwortlicher:** Prof. Dr. Thomas Herrmann

<b>Studienprojekt</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 8 CP	<b>Workload:</b> 240 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> immer	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Praktikum/Projektarbeit			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 210 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> 90 LP, siehe Prüfungsordnung</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Bitte entnehmen Sie die projektspezifischen Vorkenntnisse dem jeweiligen Studienprojekt.</p>					
<p><b>Lernziele (learning outcomes):</b> Ziele des Studienprojekts sind:</p> <ul style="list-style-type: none"> <li>• Die Anwendung des erlernten Fachwissens.</li> <li>• Der Erwerb zusätzlicher Fachkompetenz gemäß der jeweiligen projektspezifischen Aufgabenstellung.</li> <li>• Die Schulung hinsichtlich der Erarbeitung eigener Lösungsstrategien.</li> <li>• Die Schulung hinsichtlich Arbeitsteilung und Zusammenarbeit im Team (Teamfähigkeit und Projektorganisation).</li> <li>• Erwerb weiterer Kompetenz im Hinblick auf die Dokumentation und die Präsentation von Ergebnissen.</li> </ul> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Kritikfähigkeit</li> <li>• Projekt- und Zeitmanagement</li> <li>• Präsentation wissenschaftlicher Ergebnisse</li> <li>• Teamarbeit und Teamfähigkeit</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul>					
<p><b>Inhalt:</b></p> <p>Im Rahmen des Studienprojekts soll eine Aufgabe aus Bereichen der Angewandten Informatik in Teamarbeit unter Anleitung eines Betreuers gelöst werden.</p> <p>Organisation und Ablauf: Das Studienprojekt muss vor Beginn beim Prüfungsamt ange-</p>					

## KAPITEL 5. PRAKTISCHE FÄCHER

meldt werden. Bitte beachten Sie die hierfür festgelegten Fristen, die Sie unter <http://www.ai.rub.de/studierende/studienprojekte-und-abschlussarbeiten/studienprojekt.html> finden. In begründeten Fällen (Industriekooperation etc.) können Studienprojekte auch außerhalb dieses Verfahrens durchgeführt werden, wenn Sie durch den Prüfungsausschuss AI beantragt und genehmigt wurden.

Durchführungszeitraum: Studienprojekte starten mit Beginn der Vorlesungszeit und enden spätestens am 31. Juli (Sommersemester) bzw. am 15. Februar (Wintersemester) durch eine Abschlussdokumentation. Die termingerechte Abgabe und positive Bewertung durch den Dozenten sind Voraussetzung für die Anerkennung der Leistungspunkte.

**Lehrformen:**

Praktikum/Projektarbeit

**Prüfungsformen:**

Projektarbeit

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote:** 8 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Thomas Herrmann**  
Dozenten der RUB

<b>Projektmanagement</b>					
<b>Modul-Nr:</b> 260082	<b>Credits:</b> 2 CP	<b>Workload:</b> 60 h	<b>Semester:</b> 4-5 Sem.	<b>Turnus:</b> Winter- und Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Inverted Classroom Veranstaltung			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 30 h	<b>Gruppengröße:</b> 10-25
<b>Teilnahmevoraussetzungen:</b> Teilnahme an einem Studienprojekt					
<p><b>Lernziele (learning outcomes):</b> Studierende erwerben Grundkenntnisse und Fähigkeiten zum IT-Projektmanagement. Sie lernen Softwareentwicklungsprozesse, Projektabläufe und -phasen kennen sowie deren Zusammenhänge. Die theoretischen Erkenntnisse werden in Gruppen anhand ausgewählter Methoden und Instrumente des Projektmanagements an ihrem konkreten Studienprojekt vertieft.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Koordinationsfähigkeit und Aufgabenaufteilung</li> <li>• Projekt- und Zeitmanagement</li> <li>• Kritikfähigkeit</li> <li>• Präsentation von Projektergebnissen</li> <li>• Teamarbeit und Teamfähigkeit</li> <li>• Rhetorik und sprachliche Kompetenz</li> </ul>					
<p><b>Inhalt:</b> Die Studierenden beschäftigen sich mit Methoden und Werkzeugen des Projektmanagements in klassischen und agilen Projektszenarien. Dabei werden alle Projektphasen von der Initialisierung über das Controlling bis hin zum Projektabschluss behandelt. Im Fokus stehen dabei die parallel von den Studierenden bearbeiteten Bachelor-Studienprojekte und die konkrete Anwendung von Projektmanagement auf ihre eigene Arbeit. Verstärkt durch den Inverted-Classroom-Ansatz ist es den Teams möglich in eigenem Lerntempo die bereitgestellten Unterlagen und Methoden zu erlernen und in den über das Semester verteilten Präsenzterminen können dann Fragen und aktuelle Probleme besprochen und vertieft werden.</p>					
<p><b>Lehrformen:</b> Interaktive Lernfolien zum Selbstlernen. Präsenzveranstaltung zur Vertiefung der Themen und praktische Anwendung der Methoden auf die eigenen Studienprojekte. Besprechung und Austausch über aktuelle Probleme oder Hindernisse beim Management des Studienprojektes.</p>					

## KAPITEL 5. PRAKTISCHE FÄCHER

<p><b>Prüfungsformen:</b> Zwei schriftliche Hausaufgaben im Laufe des Semesters (jeweils ca. 2-5 Seiten in Gruppenarbeit). Ein Abschlussvortrag (ca. 15 Minuten).</p>
<p><b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Abgabe von zwei Hausaufgaben und aktive Teilnahme am Abschlussvortrag der Gruppe.</p>
<p><b>Stellenwert der Note für die Endnote:</b> 2 / 175</p>
<p><b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Thomas Herrmann</b> Marcel Schmittchen Msc.</p>
<p><b>Sonstige Informationen:</b> <b>Literatur:</b></p> <ol style="list-style-type: none"><li>1. Schelle, H.; Ottmann, R.; Pfeiffer, A.: ProjektManager. GPM Deutsche Gesellschaft für Projektmanagement e.V. (Hrsg.), Nürnberg, 2. Auflage 2005.</li><li>2. Tiemeyer, E. (Hrsg.): Handbuch IT-Projektmanagement. Carl Hanser Verlag, München 2010.</li><li>3. Mangold, P.: IT-Projektmanagement kompakt. Spektrum Akademischer Verlag, München 2004.</li></ol>

# Bachelorarbeit

<b>Bachelorarbeit</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 12 CP	<b>Workload:</b> 360 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> immer	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Praktikum/Projektarbeit			<b>Kontaktzeit:</b> 0 SWS	<b>Selbststudium:</b> 360 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Die Ausgabe des Themas der Bachelorarbeit kann erst dann erfolgen, wenn der Studierende mindestens 90 Leistungspunkte erreicht hat. Idealerweise sollte die Bachelor-Arbeit jedoch den Abschluss des Bachelorstudiums darstellen.</p> <p><b>Empfohlene Vorkenntnisse:</b>  Fachspezifisch von der Themenstellung abhängig.</p>					
<p><b>Lernziele (learning outcomes):</b> Die Bachelorarbeit soll zeigen, dass die oder der Studierende in der Lage ist, innerhalb einer vorgegebenen Frist eine anspruchsvolle Fragestellung der Angewandten Informatik unter Anwendung der im Bachelorstudium erworbenen Methoden selbständig zu bearbeiten. Darüber hinaus wird der Erwerb von Grundkenntnissen der wissenschaftlichen Arbeit einschließlich der Projektorganisation sowie die Präsentation der erarbeiteten Ergebnisse erwartet.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Kritikfähigkeit</li> <li>• Projekt- und Zeitmanagement</li> <li>• Präsentation wissenschaftlicher Ergebnisse</li> <li>• Rhetorik und sprachliche Kompetenz</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Wissenschaftliches Arbeiten und Schreiben</li> </ul>					

## KAPITEL 6. BACHELORARBEIT

<b>Inhalt:</b> 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 werden die Ergebnisse in Form eines Kolloquiumsvortrags präsentiert.
<b>Lehrformen:</b> Praktikum/Projektarbeit
<b>Prüfungsformen:</b> Abschlussarbeit
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 12 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Thomas Herrmann</b> Dozenten der RUB
<b>Sonstige Informationen:</b> Im Normalfall sucht sich jede bzw. jeder Studierende nach eigenem Interesse und Neigung einen Lehrstuhl aus, an dem sie bzw. er die Bachelorarbeit schreiben möchte. Die meisten Lehrstühle veröffentlichen ihre angebotenen Themen. Oft werden Themen aber auch erst nach Absprache mit dem Studierenden gestellt, wobei der Studierende ein Vorschlagsrecht hat. Die Anmeldung für die Bachelorarbeit erfolgt beim Prüfungsamt der Angewandten Informatik. Für die Anmeldung ist das persönliche Erscheinen sowie die Vorlage des aktuellen Studentenausweises bzw. der Immatrikulationsbescheinigung notwendig.

# Ingenieurinformatik

<b>Digitaltechnik</b>					
<b>Modul-Nr:</b> 141300	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <ul style="list-style-type: none"> <li>• Grundlagen der Elektronik</li> </ul> <p>Erforderlich sind zudem elementare Kenntnisse in:</p> <ul style="list-style-type: none"> <li>• Grundlagen der Elektrotechnik</li> <li>• Mathematik</li> </ul>					
<p><b>Lernziele (learning outcomes):</b> Ziel der Lehrveranstaltung 'Digitaltechnik' ist die Vermittlung elementarer Grundlagenkenntnisse aus den Bereichen Boolesche Algebra, Kostenoptimierung digitaler Schaltungen, Aufbau und die Wirkungsweisen von digitalen Grundschaltungen, Aufbau und Funktion von Basisfunktionalitäten aus denen sich z.B. ein Mikroprozessorsystem zusammensetzt (wie z.B. Zähler, Schieberegister, ALU, Bustreiber, Speicher). Weiterhin werden in der Lehrveranstaltung 'Digitaltechnik' zentrale Kenntnisse über den inneren schaltungstechnischen Aufbau aktueller Logikfamilien vermittelt, die besonderen Eigenschaften einer CMOS-Logik, die Skalierungseigenschaften von CMOS-Technologien und ihre Auswirkungen auf die elektrischen Eigenschaften logischer Schaltungen und Systeme. Mit diesem Wissen sollten die Studenten in der Lage sein, zukünftige Entwicklungen in den Integrationstechnologien, und damit in der Digitaltechnik bezüglich ihrer Möglichkeiten und Grenzen einzuschätzen.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p>					

## KAPITEL 7. INGENIEURINFORMATIK

- Fächerübergreifendes Denken und Arbeiten

Fachspezifische Kompetenzen:

- Logik und Schaltungen

### **Inhalt:**

- Historischer Rückblick, Motivation Digitaltechnik
- Boolesche Algebra
- Zahlendarstellungen, Rechenwerke, ALU
- Flankendetektoren, Flip-Flops (FFs)
- Teiler, Zähler, Schieberegister, Halbleiterspeicher
- Tools zur Logikanalyse
- Dioden-Logik, Dioden Transistor Logik, Transistor Transistor Logik, CMOS-Logik
- CMOS Technologie, Moore's Law
- CMOS Standard-Zellen Konzept

Die Vorlesung beginnt mit den theoretischen Grundlagen der Schaltalgebra. Danach werden verschiedene Verfahren zur Vereinfachung von logischen Netzwerken vorgestellt. Die vereinfachten logischen Netzwerke gilt es dann auf der Basis der schaltungstechnischen logischen Grundfunktionen NAND, NOR und NOT in kostenoptimale logische Netzwerke zu überführen. Dabei wird der Begriff der Kosten sowohl unter dem Gesichtspunkt des Hardwareaufwands, als auch unter dem Gesichtspunkt der Summe der Gatterlaufzeiten in den Signalpfaden eingeführt. Der zweite Teil der Vorlesung beschäftigt sich mit den zentralen Eigenschaften der wichtigsten Logikfamilien. Voran gestellt werden zunächst die klassischen Logikfamilien (Dioden-Logik, Dioden-Transistor-Logik, Transistor-Transistor-Logik) in Verbindung mit ihren typischen Merkmalen. Vor dem Hintergrund des aktuellen Technologiefortschritts werden daran anschließend die zentralen Merkmale einer CMOS-Technologie, das Moore'sche Gesetz, die Auswirkungen von Technologieskalierungen auf die Schaltzeiten der CMOS-Gatter, die CMOS-Logik und das CMOS-Standard-zellenkonzept vorgestellt. Der dritte Teil der Vorlesung beschäftigt sich mit den höherwertigen digitalen Funktionsgruppen. Dazu gehören z.B. Flip-flops, Zähler, Schieberegister, Multiplexer/Demultiplexer, Rechenwerke/ALU und Speicher. Die Konzepte synchroner/asynchroner Taktsteuerungen und paralleler/sequentieller Datenverarbeitung werden in Verbindung mit den möglichen unterschiedlichen Architekturen der höherwertigen Funktionsgruppen diskutiert.

### **Lehrformen:**

Vorlesung (+Übung)

### **Prüfungsformen:**

Klausur



<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Jürgen Oehm</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. Katz, Randy H. "Contemporary Logic Design", Prentice Hall, 1993 2. Seifart, Manfred, Beikirch, Helmut "Digitale Schaltungen", Verlag Technik, 1998 3. Borucki, Lorenz, Stockfisch, Georg "Digitaltechnik", Teubner Verlag, 1989 4. Pernards, Peter "Digitaltechnik I. Grundlagen, Entwurf, Schaltungen", Hüthig, 2001 5. Fricke, Klaus "Digitaltechnik. Lehr- und Übungsbuch für Elektrotechniker und Informatiker", Vieweg, 2005 6. Lipp, Hans Martin, Becker, Jürgen "Grundlagen der Digitaltechnik", Oldenbourg, 2005 7. Tietze, Ulrich, Schenk, Christoph, Gamm, Eberhard "Halbleiter - Schaltungstechnik", Springer, 2002 8. "Handbuch der Elektronik. Digitaltechnik", Medien Institut Bremen, 1999 9. Weste, Neil H. E., Eshragian, Karman, Eshragian, Kamran "Principles of CMOS VLSI Design: A Systems Perspective", Addison Wesley Longman Publishing Co, 1993 10. Wuttke, Heinz-Dieter, Henke, Karsten "Schaltssysteme. Eine automatenorientierte Einführung", Pearson Studium, 2002 11. Siemers, Christian, Sikora, Axel "Taschenbuch Digitaltechnik", Hanser Fachbuchverlag, 2002 12. Schiffmann, Wolfram, Schmitz, Robert "Technische Informatik 1. Grundlagen der digitalen Elektronik", Springer, 2003

<b>Geometrische Modellierung und Visualisierung</b>					
<b>Modul-Nr:</b> 129008	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> 1 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Objektorientierte Modellierung					
<p><b>Lernziele (learning outcomes):</b> Im Rahmen des Moduls werden die wesentlichen Grundlagen der geometrischen Modellierung sowie elementare Techniken der Visualisierung erlernt. Mit Abschluss des Moduls befinden Sie sich in der Lage, geometrische und visualisierungsbezogene Aufgaben aus Forschung und Praxis unter Verwendung aktueller Methoden der Mathematik und Information zielgerecht zu lösen.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende Techniken zur digitalen Visualisierung von Geometrie,</li> <li>• sind Studierende darin geschult räumlich zu denken,</li> <li>• verstehen Studierende wie Geometrie transformiert werden kann,</li> <li>• haben Studierende ein Verständnis über Programmierung erlernt,</li> <li>• wissen Studierende wie Geometrie persistent gespeichert werden kann,</li> <li>• sind Studierende darauf vorbereitet erlerntes auf international genutzte Anwendungen mit visueller Verarbeitung von Geometrie zu übertragen</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Geometrische Modelle</li> <li>• Affine Abbildungen und Differentialgeometrie</li> <li>• Freiformkurven und -flächen</li> <li>• Boundary Repräsentationen</li> <li>• Constructive Solid Geometry</li> <li>• Quadtrees, Octrees</li> <li>• Zerlegung und Triangulierung</li> <li>• Visualisierungstechniken</li> </ul>					

## KAPITEL 7. INGENIEURINFORMATIK

<b>Lehrformen:</b> Vorlesungen, Übungen am Computer
<b>Prüfungsformen:</b> Klausur, Dauer: 180 min
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Verwendung des Moduls (in anderen Studiengängen):</b> Angewandte Informatik (Bachelor); Bauingenieurwesen (Master)
<b>Stellenwert der Note für die Endnote:</b> 6 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr.-Ing. Markus König</b> Andre Vonthron, M.Sc. Marcel Stepien, M. Sc.
<b>Sonstige Informationen:</b> <b>Literatur:</b>  1. Gerald Farin: Curves and Surfaces for CAGD - A Practical Guide. 5th Edition. San Francisco: Morgan Kaufmann. 2002  2. David Salomon: Curves and Surfaces for Computer Graphics. New York: Springer. 2006  3. Raymond Zavodnik, Herbert Kopp: Graphische Datenverarbeitung - Grundzüge und Anwendungen. München u.a.: Carl Hanser

<b>Grundlagen der Automatisierungstechnik</b>					
<b>Modul-Nr:</b> 135110	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Grundlagen der Automatisierungstechnik			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppengröße:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Zielsetzung:</p> <ul style="list-style-type: none"> <li>• 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.</li> <li>• 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.</li> <li>• 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.</li> </ul> <p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Die Studierenden kennen wesentliche Methoden und Verfahren der Ingenieurwissenschaften / des Maschinenbaus, verfügen über entsprechendes Fachvokabular und kennen Anwendungsbeispiele.</li> <li>• Die Studierenden kennen vertiefte ingenieurwissenschaftliche Grundlagen im Bereich ihres Studienschwerpunkts.</li> </ul> <p>Fertigkeiten:</p> <ul style="list-style-type: none"> <li>• Die Studierenden verfügen über die Fähigkeit zu vernetztem und kritischem Denken.</li> <li>• Die Studierenden praktizieren erste Ansätze wissenschaftlichen Lernens und Denkens.</li> <li>• Die Studierenden können ingenieurtechnische Probleme modellieren und lösen.</li> <li>• Die Studierenden können komplexe mathematische Problemstellungen in physikalischen Systemen (ggf. fachübergreifend) mit geeigneten Methoden lösen.</li> </ul>					

## KAPITEL 7. INGENIEURINFORMATIK

- Die Studierenden haben die Fähigkeit zu vernetztem und kritischem Denken ausgebaut und sind in der Lage etablierte Methoden und Verfahren auszuwählen und anzuwenden.

Kompetenzen:

- Die Studierenden verfügen über fachübergreifende Methodenkompetenz.
- Die Studierenden können Erkenntnisse / Fertigkeiten auf konkrete maschinenbauliche / ingenieurwissenschaftliche Problemstellungen übertragen.

### **Inhalt:**

Nach einem allgemeinen historischen Überblick über die Entwicklung der Automatisierungstechnik werden wesentliche Entwicklungsmethoden und Notationen für Automatisierungsaufgaben vorgestellt. Ein Schwerpunkt der Lehrveranstaltung steht die SPS mit ihrem Hardwareaufbau und dem Echtzeitbetriebssystem. Die SPS-Programmierung wird in Laborübungen vertieft. Dabei spielt die Signalverarbeitung von der Erfassung der Sensorsignale über die Verarbeitung im Steuerungsalgorithmus bis zur Ausgabe der Steuerbefehle an die Stellglieder eine wesentliche Rolle. Die Anwendung des PC für die industrielle Automatisierung und die dezentrale Signalerfassung und -ausgabe werden exemplarisch behandelt. Einen weiteren Schwerpunkt 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 sowohl Robotersteuerungen als auch numerischen Steuerungen werden in Vorlesungs- und Übungseinheiten vertieft. Die Lehrveranstaltung schließt mit einer Einführung in die Grundlagen der Kommunikationstechnik, Sensorik und Sicherheitstechnik im Themenfeld der Automatisierung ab. 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:**

Vorlesungseinheiten und Übungseinheiten zur Vertiefung der Vorlesungsinhalte; In den Vorlesungseinheiten werden ebenfalls kleinere Übungsaufgaben (zum Teil in Gruppenarbeiten) durchgeführt.

### **Prüfungsformen:**

Schriftliche Prüfung (180 Minuten) in der Prüfungszeit nach Vorlesungsende

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

### **Verwendung des Moduls (in anderen Studiengängen):**

Studiengang Maschinenbau, Studiengang SEPM, Studiengang Angewandte Informatik, Studiengang Elektrotechnik

**Stellenwert der Note für die Endnote:** 6 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr.-Ing. Bernd Kuhlenkötter**

## KAPITEL 7. INGENIEURINFORMATIK

Vorlesungsbetreuer: Thomas Glaser, M. Sc.; Paul Glogowski, M. Sc. Übungsbetreuer: Dennis Möllensiep, M. Sc.; Michael Krampe, M. Sc.; Mathias Weißköppel, B. Sc.; Stefanie Spies, M. Sc.;

### **Sonstige Informationen:**

#### **Literatur:**

1. Abel, D.: Petri-Netze für Ingenieure - Modellbildung und Analyse diskret gesteuerter Systeme, Springer-Verlag, Berlin, 1990.
2. John, K.-H., Tiegelkamp, M.: SPS-Programmierung mit IEC 1131-3, Springer Verlag, Berlin, 2009.
3. Stein, G.: Automatisierungstechnik in der Maschinenteknik, Hanser Verlag, München, 1994.
4. Weck, M.: Werkzeugmaschinen Fertigungssysteme, VDI Verlag, 2006.
5. Wellenreuther, G.; Zastrow, D.: Steuerungstechnik mit SPS
6. Vieweg Verlag, Wiesbaden, 1998.
7. Wloka, Dieter W.: Robotersysteme Band 1: Technische Grundlagen
8. Springer Verlag, Berlin, 1992.

DIN-Normen und Richtlinien

<b>Mensch-Maschine-Interaktion</b>					
<b>Modul-Nr:</b> 260083	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung und Übung integriert			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 75 h	<b>Gruppen- größe:</b> 15-25
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden lernen, was bei der Gestaltung von Benutzungsschnittstellen heute und auch bei zukünftigen Formen der Mensch-Maschine-Interaktion zu berücksichtigen ist, und wie man beim Entwicklungsprozess vorgeht.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Entwicklungsprozesse</li> </ul>					
<p><b>Inhalt:</b></p> <ol style="list-style-type: none"> <li>1. Relevante Eigenschaften des Menschen und seines Arbeitskontextes - Physiologie - Wahrnehmungspsychologie - Kognitionspsychologie - Arbeitspsychologie</li> <li>2. Usability-Engineering - Methoden der ergonomischen Gestaltung der Mensch-Computer-Schnittstelle <ul style="list-style-type: none"> <li>• Methoden der Anforderungsanalyse</li> <li>• Maskengestaltung</li> <li>• Prototyping</li> <li>• Usability-Testing</li> </ul> </li> <li>3. Die Nutzung computerbasierter Kommunikations- und Informationssysteme wird in der Regel in menschliche Handlungsabläufe eingebettet. Die Vorlesung vermittelt Kenntnisse derjenigen Faktoren und Methoden, die bei der Entwicklung und Einführung informationstechnischer Systeme dazu beitragen, dass die Nutzung erfolgreich ist. Dabei wird davon ausgegangen, dass technische, organisatorische und soziale Strukturen integriert und angepasst werden müssen. Die Erfolgsfaktoren werden aus interdisziplinärer Sicht behandelt und anhand von Beispielen aus konkreten Anwendungsfällen erläutert. Dabei werden die folgenden Gebiete berücksichtigt: <ul style="list-style-type: none"> <li>• Arbeitswissenschaft und Ergonomie</li> </ul> </li> </ol>					

## KAPITEL 7. INGENIEURINFORMATIK

- Psychologie
- Organisation und Management
- Rechtliche Aspekte
- Betriebswirtschaftliche Aspekte
- Kommunikationstheorie
- Datenschutz

4. Es werden verschiedene Methoden vermittelt, die die Einführung von Informationssystemen unterstützen:

- Erhebung von Anforderungen und Ausgangsbedingungen
- Usability-Engineering
- Contextual Design
- Sozio-technische Modelleirung
- Partizipation und Kommunikationsmoderation

### **Lehrformen:**

Abwechslung von Vorlesungsinhalten, Seminarbeiträgen der Studierenden und Präsenzübungen

### **Prüfungsformen:**

Seminarbeitrag und Klausur

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Seminarbeitrag (50% der Endnote) und Klausur (60 Minuten, 50% der Endnote)

**Stellenwert der Note für die Endnote:** 5 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Thomas Herrmann**

Markus Jelonek Msc.

### **Sonstige Informationen:**

Im Wintersemester 2019/20 kann die Vorlesung "Mensch-Maschine Interaktion" nicht von Prof. Herrmann angeboten werden. Er hat mit Prof. Schneegaß an der Universität Duisburg-Essen (Campus Essen) vereinbart, dass die dort durchgeführte Veranstaltung Human-Computer Interaction ersatzweise von maximal 14 Studierenden wahrgenommen werden kann:

<https://www.hci.wiwi.uni-due.de/aktuelles/einzelansicht/vorlesungen-ws-20192019893/>

Die Anmeldung kann bis zum 20. September 2019 erfolgen. Die Reihenfolge der Anmeldung ist ausschlaggebend. Melden Sie sich bitte beim Lehrstuhl IMTM über [marina.konrad@ruhr-uni-bochum.de](mailto:marina.konrad@ruhr-uni-bochum.de) an. Zusätzlich ist die Anmeldung zur Prüfung über Flexnow innerhalb der Anmeldephase vorzunehmen.



**Literatur:**

1. Bernhard Preim, R.D. (2010). Interaktive Systeme. Springer-Verlag Berlin-Heidelberg.
2. Herzeg, M. (2005). Software-Ergonomie. Oldenbourg Wissenschaftsverlag.

<b>Virtuelle Produktmodellierung und Visualisierung</b>					
<b>Modul-Nr:</b> 135060	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Virtuelle Produktmodellierung und Visualisierung			<b>Kontaktzeit:</b> V: 2 SWS; Ü: 2 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppengröße:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• verfügen Studierende über ein breites, integriertes Wissen über die Herausforderungen moderner Produktentstehungsprozesse und die resultierenden Anforderungen an Softwaresysteme zur Virtuellen Produktmodellierung und -visualisierung.</li> <li>• 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.</li> <li>• 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.</li> <li>• können Studierende Aufgabenstellungen der Virtuellen Produktmodellierung und -visualisierung reflektieren und bewerten sowie selbstgesteuert verfolgen</li> <li>• können Studierende kooperativ Aufgabenstellungen der Virtuellen Produktmodellierung und -visualisierung in heterogenen Gruppen bearbeiten, Abläufe und Ergebnisse begründen sowie über Sachverhalte umfassend kommunizieren</li> </ul>					
<p><b>Inhalt:</b> Die Veranstaltung 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</p>					

## KAPITEL 7. INGENIEURINFORMATIK

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.
<b>Lehrformen:</b> Vorlesungen, Übungen, Gruppenarbeiten, Moodle Unterstützung
<b>Prüfungsformen:</b> <ul style="list-style-type: none"><li>• Schriftliche Klausur mit Fragen und Aufgaben zum Stoff der Vorlesung und der Übung</li><li>• Dauer: 90 Minuten, Anteil der Modulnote: 100%</li></ul>
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulklausur Prüfungsvorleistung: Im Rahmen der Übung werden 5 Projektaufgaben gestellt, die bearbeitet und abgegeben werden müssen. Für die Zulassung zur Modulklausur müssen mind. 4 Projektaufgaben positiv bewertet sein
<b>Verwendung des Moduls (in anderen Studiengängen):</b> Studiengang Maschinenbau (Bachelor), Studiengang SEPM (Bachelor)
<b>Stellenwert der Note für die Endnote:</b> 6 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr.-Ing. Detlef Gerhard</b>

<b>Umweltinformatik</b>					
<b>Modul-Nr:</b> 127018	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1/2 Semester
<b>Lehrveranstaltungen:</b> Umweltinformatik			<b>Kontaktzeit:</b> V: 1 SWS; Ü: 1 SWS	<b>Selbststudium:</b> 60 h (Vor- und Nachbereitung inkl. Prüfung) h	<b>Gruppengröße:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Grundkenntnisse in der Programmierung mit MATLAB werden erwünscht und sind vor der Teilnahme an den Computerübungen selbstständig nachzuarbeiten.					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden erlernen spezielle Methoden zur Modellierung und Analyse von komplexen Umweltsystemen. Im Fokus stehen dabei insbesondere Konzepte zur Modellierung von Umweltdaten sowie geometrische Algorithmen und räumliche Analysen in Punktmengen. Darüber hinaus werden praktische Problemstellungen mittels Graphen modelliert und durch die Anwendung von bekannten Algorithmen aus den Bereichen Wegsuche und Flusstheorie analysiert. Zusätzlich dazu werden im Modul gängige Interpolationsverfahren für Umweltdaten vorgestellt.</p> <p>Nach erfolgreichem Abschluss des Moduls,</p> <ul style="list-style-type: none"> <li>• kennen Studierende grundlegende Methoden zur Abbildung von Umweltdaten.</li> <li>• können Studierende Umweltdaten abbilden und analysieren.</li> <li>• verstehen Studierende auf Basis der vorgestellten Algorithmen, wie gängige Verfahren zur Geodatenanalyse funktionieren.</li> <li>• können Studierende die vorgestellten Algorithmen auf weitere Problemstellungen übertragen.</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Grundlagen der Graphentheorie</li> <li>• Modellierung von Geodaten in Graphen</li> <li>• Anwendung von Graphenalgorithmen: Wegsuche</li> <li>• Anwendung von Graphenalgorithmen: Netzwerkanalysen</li> <li>• Räumliche Analysen und Zerlegung von Punktmengen</li> <li>• Interpolationsverfahren</li> </ul>					
<b>Lehrformen:</b>					

## KAPITEL 7. INGENIEURINFORMATIK

Vorlesung, Computerübung in Gruppen
<b>Prüfungsformen:</b> schriftliche Modulabschlussprüfung (Klausur) von 75 min
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Verwendung des Moduls (in anderen Studiengängen):</b> Studiengang Umwelttechnik und Ressourcenmanagement (M.Sc.)
<b>Stellenwert der Note für die Endnote:</b> 3 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  Prof. Dr. Markus König

<b>Fertigungsautomatisierung</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Keine</p>					
<p><b>Lernziele (learning outcomes):</b> Aufbauend auf den im Bachelor-Studiengang vermittelten Kenntnissen über Automatisierungstechnik lernen die Studierenden vertiefte ingenieurwissenschaftliche Grundlagen im Bereich der automatisierten Fertigungsverfahren mit bahngesteuerten Werkzeugen kennen. Ein Schwerpunkt wird dabei auf die NC- und Robotersteuerungen und deren Programmierung gelegt. Ein zweiter Schwerpunkt befasst sich mit vernetzten Fertigungssystemen.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen: Technologien:</p>					
<p><b>Inhalt:</b></p> <p>In der Veranstaltung werden zunächst unterschiedliche Fertigungsverfahren vorgestellt, um deren spezifischen Anforderungen an die Automatisierung herauszuarbeiten. Im Fokus stehen hierbei innovative Fertigungsverfahren wie das Rapid-Prototyping, die Hochgeschwindigkeitszerspanung, die inkrementelle Umformung oder die Laserbearbeitung. Im Abschnitt NC-Steuerungen werden die Datenaufbereitung, die Bahnsteuerungsfunktionen mit Geschwindigkeitsführung, Interpolation und Koordinatentransformation sowie die Lageregelung behandelt. Es werden Entwicklungspotentiale in Richtung offene NC-Steuerungen und STEP-NC aufgezeigt. In Abschnitt Robotersteuerungen werden insbesondere die spezifischen Probleme und Lösungen der Transformation vom Effektor- zum Basiskoordinatensystem vorgestellt. Die für Werkzeugmaschinen und Roboter wichtigen Wegmesssysteme werden in ihrer Funktionsweise erläutert. Ein weiterer Abschnitt behandelt das Thema Genauigkeit und stellt die für NC-Werkzeugmaschinen und Roboter zu berücksichtigenden Normen vor. Die wichtigen Feldbusse PROFIBUS und INTERBUS sowie die Sensor-/Aktorbusse CAN und SERCOS werden in Aufbau und Kommunikationsstruktur eingehend vermittelt und die Potentiale der Internettechnik in Steuerungsanwendungen behandelt. Im Abschnitt sicherheitsgerichtete Steue-</p>					

## KAPITEL 7. INGENIEURINFORMATIK

rungen werden die relevanten Konzepte für SPS- Sicherheitssteuerungen und sichere Feld- und Sensor- Aktorbusse dargestellt.
<b>Lehrformen:</b> Vorlesung (+Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 6 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  Prof. Dr.-Ing. Bernd Kuhlenkötter

# Bioinformatik

<b>Grundlagen der Bioinformatik</b>					
<b>Modul-Nr:</b> 190533 (Übung: 190543)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<b>Empfohlene Vorkenntnisse:</b>  Grundkenntnisse in Biologie					
<b>Lernziele (learning outcomes):</b> 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.					
<b>Vermittelte Kompetenzen:</b> Kernkompetenzen:					
<ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul>					
<b>Inhalt:</b> 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.					



## KAPITEL 8. BIOINFORMATIK

<b>Lehrformen:</b> Vorlesung (+Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>PD Dr. Mathias Lübben, Prof. Dr. Axel Mosig</b> PD Dr. Mathias Lübben, Prof. Dr. Axel Mosig, Prof. Dr. Raphael Stoll
<b>Sonstige Informationen:</b> <b>Literatur:</b>  A. Lesk, Introduction to Bioinformatics, Oxford University Press, 2002.  Biologische Grundkenntnisse können ggf. durch entsprechende Online-Kurse erworben werden.

<b>Methoden der Bioinformatik</b>					
<b>Modul-Nr:</b> 190502 (Übung: 190522)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Grundkenntnisse in Biologie</p>					
<p><b>Lernziele (learning outcomes):</b> 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).</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Selbständiges Lernen und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Graphentheorie</li> <li>• Mathematik</li> <li>• Optimierungsmethoden</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Algorithmische und statistische Grundlagen der Bioinformatik (Reguläre Ausdrücke, Endliche Automaten, Turing Maschinen, Komplexität, Dynamische Programmierung, Maximum Likelihood, Hidden Markov Modelle, Poisson Prozesse)</li> <li>• Algorithmen zur Analyse von Sequenz und Struktur von Bio-Molekülen; Rekonstruktion evolutionärer Beziehungen zwischen Sequenzen und Strukturen</li> </ul>					

## KAPITEL 8. BIOINFORMATIK

<b>Lehrformen:</b> Vorlesung (+Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Axel Mosig</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b> <ol style="list-style-type: none"><li>1. R. Durbin, S. Eddy, A. Krogh, G. Mitchinson, Biological Sequence Alignments, Cambridge University Press, 2004.</li><li>2. N. Jones, P. Pevzner, An Introduction to Bioinformatics Algorithms, MIT Press, 2004.</li><li>3. P. Pevzner, R. Shamir, Computing Has Changed Biology—Biology Education Must Catch Up, Science 325(5940):541-542, 2009.</li><li>4. T.W. Tan, S.J. Lim, A.M. Khan, S. Ranganathan, A proposed minimum skill set for university graduates to meet the informatics needs and challenges of the "-omics" era, BMC Genomics 10(Suppl 3):S36, 2009.</li></ol> <p>Biologische Grundkenntnisse können ggf. durch entsprechende Online-Kurse erworben werden.</p>

# Computerlinguistik

<b>Computerlinguistik 1</b>					
<b>Modul-Nr:</b>	<b>Credits:</b>	<b>Workload:</b>	<b>Semester:</b>	<b>Turnus:</b>	<b>Dauer:</b>
CL1	5 CP	150 h	3-6 Sem.	Sommersemester	1 Semester
<b>Lehrveranstaltungen:</b> Das Modul besteht aus zwei Teilen, einem Kurs im Seminarstil mit Interaktion und einer Übung, in der u. a. praktische Übungen zu den im Kurs besprochenen Inhalten durchgeführt werden und schwierige Elemente des Lernstoffs nachbereitet werden.			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> ca. 10-15 Studierende
<b>Teilnahmevoraussetzungen:</b> Erfolgreiche Teilnahme am Modul Einführung in die Linguistik.					
<b>Empfohlene Vorkenntnisse:</b>  Grundlagen der Sprachwissenschaft, wie sie im Kurs 'Einführung in die Linguistik' vermittelt werden.					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls verfügen die Studierenden über Kenntnisse in klassischen Methoden der Computerlinguistik. Sie kennen die wichtigsten Verfahren für die morphologische, syntaktische und semantische Analyse (Automatentheoretische Konzepte, Verwendung von Transduktoren, Parsingverfahren, kompositionelle sowie distributionelle Semantik) und können die Vor- und Nachteile der verschiedenen Verfahren bei der konkreten linguistischen Analyse einschätzen. Die Studierenden können die Ansätze implementieren (vorzugsweise in der Programmiersprache Python) und die besprochenen Verfahren auf eigene Fragestellungen anwenden.					
<b>Inhalt:</b> Gegenstand des Moduls sind symbolische Verfahren in der Computerlinguistik. Hierfür werden zunächst die notwendigen Konzepte aus der theoretischen Informatik, die für die morphologische und syntaktische Analyse relevant sind, vorgestellt (Reguläre Sprachen und Relationen, Transduktoren; kontextfreie Grammatiken, Parsingverfahren: Shift-Reduce Parsing, Earley-Algorithmus, CYK-Parsing, außerdem probabilistisches Parsing). Dependenzparsing als Beispiel für korpusorientiertes Parsing. Für die semantische Verarbeitung werden die Arbeit					

## KAPITEL 9. COMPUTERLINGUISTIK

mit dem Lambda-Kalkül, logische Analysen, sowie Vektorraum-basierte lexikalische Analysen für die Lesartendisambiguierung vorgestellt. Für sämtliche dieser Inhalte werden beispielhafte Implementierungen in Python vorgestellt.

### **Lehrformen:**

Das Modul besteht aus zwei Teilen, einem Kurs im Seminarstil mit Interaktion und einer Übung, in der u. a. praktische Übungen zu den im Kurs besprochenen Inhalten durchgeführt werden und schwierige Elemente des Lernstoffs nachbereitet werden.

### **Prüfungsformen:**

Die Prüfungsformen sind

- Drei Programmieraufgaben (Bearbeitungszeit von jeweils 4 Wochen)
- Präsentation

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Sämtliche Programmieraufgaben wurden erfolgreich bearbeitet, eine Präsentation im Rahmen der Übungen

### **Verwendung des Moduls (in anderen Studiengängen):**

Studiengang Linguistik (BA)

**Stellenwert der Note für die Endnote:** 5 / 175

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Ralf Klabunde**

### **Sonstige Informationen:**

#### **Literatur:**

1. Friedl, J. (2008). Reguläre Ausdrücke (3rd ed.). Köln, DE: O'Reilly.
2. Lutz, M. (2011). Programming Python (4th ed.). Beijing, CN: O'Reilly.
3. Pilgrim, M. (2009). Dive into Python 3.
4. <http://getpython3.com/diveintopython3>
5. van Rossum, G. et al. (2012). Python documentation.
6. <http://docs.python.org>

<b>Einführung in die Linguistik</b>					
<b>Modul-Nr:</b> 050005	<b>Credits:</b> 8 CP	<b>Workload:</b> 240 h a) 75 h b) 75 h c) 60 h d) 30 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> a), b): WS c), d): SoS	<b>Dauer:</b> 2 Semester
<b>Lehrveranstaltungen:</b> a) Einführung in die Syntax b) Einführung in die Semantik c) Linguistische Methoden			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> ca. 170 h h	<b>Gruppen- größe:</b> 80-100 Studierende (inkl. Studierende anderer Fächer)
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Dieses Modul vermittelt fachspezifische Grundlagen in den Kernbereichen der Linguistik sowie fachübergreifende generische Kompetenzen im Bereich wissenschaftlichen Arbeitens. Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende die grundlegenden Begriffe und Konzepte der Syntax und Semantik</li> <li>• haben die Studierenden einen grundlegenden Überblick über die Kernbereiche der Syntax und Semantik</li> <li>• verstehen die Studierenden die elementaren analytischen Methoden der strukturellen Linguistik in den Bereichen Syntax und Semantik und können diese anwenden</li> <li>• entwickeln die Studierenden ein Verständnis für eine wissenschaftliche Auseinandersetzung mit grammatischen Strukturen und können (digitalisierte) sprachliche Daten bezüglich ihrer grammatischen Struktur analysieren sowie den Bezug zwischen Syntax und Semantik herstellen (Kompositionalität)</li> <li>• wissen Studierende, in welchen Formen wissenschaftliche Ergebnisse und Erkenntnisse in der Linguistik, Computerlinguistik und angrenzenden Fächern, z. B. Informatik und Psychologie, publiziert werden</li> <li>• können Studierende Instrumente zur Literaturrecherche nutzen, insbesondere digitale Rechercheinstrumente auf Datenbanken</li> <li>• sind die Studierenden in der Lage, wissenschaftliche Veröffentlichungen zu einem ihnen vertrauten Thema inhaltlich zu vernetzen und mündlich und schriftlich fachgerecht zu referieren</li> <li>• verfügen die Studierenden über grundlegende Fähigkeiten zur Entwicklung experimenteller Untersuchungen als ein Beispiel empirischen Arbeitens.</li> </ul>					

- können die Studierenden eigene empirische Untersuchungen für wissenschaftliche Vorträge und schriftliche Arbeiten ziel- und anlassbezogen aufbereiten und dokumentieren.

**Inhalt:**

Im Modul Einführung in die Linguistik geht es um die Vermittlung linguistischer Grundlagen in den Bereichen Syntax und Semantik, der zugehörigen syntaktischen und semantischen Analysemethoden sowie fachübergreifenden Grundlagen zum wissenschaftlichen Arbeiten:

- Syntax beschäftigt sich mit der Kombination von Wörtern zu komplexeren Einheiten bis hin zu Sätzen. Sie untersucht dabei, welche Wörter mit welchen anderen kombiniert werden können, welche Stellungsregularitäten dabei auftreten und wie die resultierende Struktur mit der Bedeutung solcher Einheiten zusammenhängt.
- Semantik ist das Teilgebiet der Linguistik, das sich mit den Bedeutungen sprachlicher Ausdrücke beschäftigt. Ein zentrales Thema ist folglich die Beantwortung der Frage, was man unter Bedeutung eines sprachlichen Ausdrucks“ versteht. Dabei lernen Sie grundlegende Bedeutungsarten kennen und beschäftigen sich mit den Bedeutungen von Wörtern, sprich der lexikalischen Semantik. Ferner bekommen Sie einen Überblick über Bedeutungsphänomene auf Satzebene. Hier werden Verfahren für eine kompositionelle Semantikonstruktion vermittelt, insbesondere die Analyse mit Hilfe der Prädikatenlogik und des Lambda-Kalküls, aber auch die nichtkompositionellen Aspekte der Semantik werden thematisiert.
- Im Kurs Linguistische Methoden werden fachübergreifende linguistische Analysemethoden thematisiert sowie Grundlagen empirisch-experimentellen Arbeitens – von der Erarbeitung des Forschungsstandes und der Entwicklung der Fragestellung über die Operationalisierung und die Formulierung von Hypothesen, die Datenerhebung, die deskriptive und inferenzstatistische Auswertung der Daten und die Bewertung der Ergebnisse bis zur Dokumentation der Untersuchung. In diesem Zusammenhang treten die Recherche, Rezeption und Aufbereitung wissenschaftlicher Fachtexte sowie Grundlagen zu wissenschaftlichem Schreiben hinzu. Dazu gehört das Einüben eines angemessenen Umgangs mit geistigem Eigentum sowie die Vermittlung von Kenntnissen über die Mechanismen der Qualitätssicherung bei wissenschaftlichen Veröffentlichungen, bspw. über Begutachtungsverfahren, und ihrer Bewertung bei der eigenen Literaturrecherche – Kenntnisse hierüber sind angesichts der fast flächendeckenden Verfügbarkeit wissenschaftlicher Texte in digitaler Form und der fortschreitenden Nutzung von OpenScience-Formaten in der Wissenschaft dringlicher denn je.

**Lehrformen:**

Vorlesungseinheiten mit begleitenden Übungen in Einzelarbeit, projektbezogene Arbeit in Kleingruppen

**Prüfungsformen:**

schriftliche Modulabschlussprüfung (schriftliche Ausarbeitung im Umfang von 800 bis 1000 Wörtern), Bearbeitungszeit 2-3 Wochen

**Voraussetzungen für die Vergabe von Kreditpunkten:**

bestandene Studienleistungen in Einführung in die Syntax (Klausur), Einführung in die Semantik (Klausur) und Linguistische Methoden (Vortragsausarbeitung), bestandene Modulabschlussprüfung

**Verwendung des Moduls (in anderen Studiengängen):**

Die in diesem Modul zusammenfassten Kurse sind im B.A.-Studiengang Linguistik (B.A.) anderen Modulen zugeordnet.

**Stellenwert der Note für die Endnote:** 8 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Stefanie Dipper, Prof. Dr. Ralf Klabunde, Jun.-Prof. Dr. Agata Renans**

**Sonstige Informationen:**

Die im Modul verwendete Literatur ist größtenteils englischsprachig; die Studierenden werden so systematisch an das Englische als Wissenschaftssprache herangeführt.

Das Modul setzt sich zusammen aus den Veranstaltungen Einführung in die Semantik und Einführung in die Syntax. Für Studierende der Angewandten Informatik wird aber eine gemeinsame Abschlussklausur gestellt. Es müssen also beide Veranstaltungen zeitgleich belegt werden.

**Literatur:**

Wird im Kurs bekannt gegeben



<b>Computerlinguistik 2</b>					
<b>Modul-Nr:</b> CL2	<b>Credits:</b> 3 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Kurs "Symbolische und statistische Verfahren" mit Übung			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> ca. 10-15 Studierende
<b>Teilnahmevoraussetzungen:</b> Erfolgreiche Teilnahme am Modul Einführung in die Linguistik sowie am Modul Computerlinguistik 1					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls verfügen die Studierenden über vertiefte Kenntnisse in klassischen und modernen Methoden der Computerlinguistik. Sie kennen die wichtigsten, grundlegenden Analyseebenen in der Computerlinguistik (wie Token- und Satzgrenzen, Wortarten, Lemmata, Word Senses, syntaktische Parses), verschiedene symbolische und statistische Verfahren zur automatischen Analyse von Sprachdaten, insbesondere verschiedene statistische Modelle zur Repräsentation sprachlichen Wissens und die dazugehörigen effizienten Algorithmen, um die Modelle anzuwenden. Sie kennen insbesondere die spezifischen Probleme bei der automatischen Verarbeitung sprachlicher Daten (wie z. B. sparse data, Ambiguitäten) und die wichtigsten Smoothing-Verfahren. Zudem kennen sie eine Reihe von Evaluationsverfahren und deren Vor- und Nachteile sowie relevante Ressourcen (Standard-Tagsets, getaggte Korpora, Baumbanken). Die Studierenden können verschiedene, auch komplexere Ansätze nachimplementieren, die besprochenen Verfahren auf eigene Fragestellungen anwenden, den Output eines vorliegenden oder ihres eigenen Programms geeignet evaluieren. Sie können komplexere Programme in einer modernen Programmiersprache wie Python erstellen.					
<b>Inhalt:</b> Gegenstand des Moduls sind symbolische und statistische Verfahren in der Computerlinguistik. Während im Modul Computerlinguistik 1 (Computerlinguistische Programmierung) der Schwerpunkt auf symbolischen Verfahren in den Kernbereichen der Linguistik (Morphologie, Syntax und Semantik) lag, beschäftigen sich die Studierenden im Modul Computerlinguistik 2 (Symbolische und statistische Verfahren) darüber hinaus mit anwendungsnahen Themenbereichen, wie z. B. der Datenaufbereitung für die weitere Analyse oder effizienten Algorithmen in der Anwendung. Außerdem werden vorrangig statistische Verfahren behandelt, bei denen das System Informationen aus Daten lernt. Neben klassischen probabilistischen Verfahren werden auch Methoden des Deep Learning behandelt, das in den letzten Jahren vermehrt Anwendung in der maschinellen Sprachverarbeitung findet. Kursthemen sind z. B. Tokenisierung und Satzgrenzenerkennung, Sprachmodelle, Kollokationsextraktion, Wortartenannotation (POS-Tagging), syntaktisches Parsing, Wortbedeutungsdisambiguierung.					
<b>Lehrformen:</b> Das Modul besteht aus zwei Teilen, einem Kurs im Seminarstil mit intensiver Interaktion und einer Übung, in der u. a. praktische Übungen zu den im Kurs besprochenen Inhalten					

## KAPITEL 9. COMPUTERLINGUISTIK

durchgeführt werden, die schriftlichen Hausaufgaben besprochen und schwierige Elemente des Lernstoffs nachbereitet werden.

### **Prüfungsformen:**

Die Prüfungsformen sind

- Zwei Programmieraufgaben (als schriftliche Hausaufgaben mit einer Bearbeitungszeit von 3 Wochen)
- Präsentation
- Klausur von 45 min

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Zwei bestandene schriftliche Hausaufgaben (Programmieraufgaben), eine erfolgreiche Präsentation im Rahmen der Übungen, eine bestandene Modulklausur am Semesterende

### **Verwendung des Moduls (in anderen Studiengängen):**

Studiengang Linguistik (BA)

**Stellenwert der Note für die Endnote: 3 / 175**

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Stefanie Dipper**

<b>Symbolische und statistische Verfahren</b>					
<b>Modul-Nr:</b> 050008	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung und Übung			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b>                      Programmierkenntnisse in Python, wie sie im Kurs 'CL-Program-mierung' vermittelt werden (u.a. objektorientierte Programmierung, vertiefte Kenntnis regulärer Ausdrücke, rekursive Funktionen).</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Grundlagen der Sprachwissenschaft, wie sie im Kurs 'Einführung in die Linguistik' vermittelt werden.</p>					
<p><b>Lernziele (learning outcomes):</b>                      Der Erwerb grundlegender Kenntnisse über symbolische und statistische Verfahren und ihre praktische Implementierung innerhalb computerlinguistischer Anwendungen.</p> <p><b>Vermittelte Kompetenzen:</b></p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Mathematik</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Stochastische Grundlagen</li> <li>• Tokenisierung</li> <li>• Satzgrenzenerkennung</li> <li>• Sprachmodelle</li> <li>• Kollokationsextraktion</li> <li>• Wortartenannotation (POS-Tagging)</li> <li>• Syntaktisches Parsing</li> <li>• Wortbedeutungsdisambiguierung</li> </ul>					

## KAPITEL 9. COMPUTERLINGUISTIK

<b>Lehrformen:</b> Vorlesung und Übung
<b>Prüfungsformen:</b> Übungsaufgaben und Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Stephanie Dipper</b> Prof. Dr. phil. Stefanie Dipper
<b>Sonstige Informationen:</b> Teilnahme ist nur nach vorheriger Anmeldung in CampusOffice möglich. Diese Anmeldung muss im WS bis zum 01.10. und im SS bis zum 01.04. erfolgen. Zusätzlich ist die Anmeldung zur Prüfung über FlexNow innerhalb der gültigen Anmeldephase notwendig.  <b>Literatur:</b>  1. Carstensen, K.-U., Ebert, C., Ebert, Ch., Jekat, S., Klabunde, R., und Langer, H. (Eds.) (2010). Computerlinguistik und Sprachtechnologie: Eine Einführung (3rd ed.). Heidelberg, DE: Spektrum.  2. Jurafsky, D., und Martin, J. H. (2009). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (2nd ed.). Upper Saddle River, NJ: Prentice-Hall.  3. Manning, C. D., und Schütze, H. (1999). Foundations of statistical natural language processing. Cambridge, MA: MIT Press.

# Kryptologie

<b>Diskrete Mathematik II</b>					
<b>Modul-Nr:</b> 150310 (Übung: 150311)	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 135 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine <b>Empfohlene Vorkenntnisse:</b>  Keine					
<p><b>Lernziele (learning outcomes):</b> Die Vorlesung gibt eine Einführung in die Codierungstheorie und in die Theorie der Berechenbarkeit.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Graphentheorie</li> <li>• Mathematik</li> <li>• Optimierungsmethoden</li> </ul>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Eindeutig entschlüsselbare Codes</li> <li>• Kompakte und optimale Codes</li> <li>• Lineare und duale Codes</li> </ul>					

## KAPITEL 10. KRYPTOLOGIE

- Turingmaschine
- Komplexitätsklassen P und NP
- Polynomielle Reduktion
- Quadratische Reste

**Lehrformen:**

Vorlesung (+Übung)

**Prüfungsformen:**

Klausur

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote: 6 / 175**

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Alexander May**

**Sonstige Informationen:**

Im Studiengang ITS läuft die Vorlesung unter dem Titel "Einführung in die theoretische Informatik".

<b>Zahlentheorie</b>					
<b>Modul-Nr:</b> 150232 (Übung: 150233)	<b>Credits:</b> 9 CP	<b>Workload:</b> 270 h	<b>Semester:</b> 3-5 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 6 SWS	<b>Selbststudium:</b> 180 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine <b>Empfohlene Vorkenntnisse:</b>  Lineare Algebra					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> <li>• kennen Studierende typische Fragestellungen der elementaren Zahlentheorie,</li> <li>• haben Studierende einen Einblick in die Ästhetik der ganzen und rationalen Zahlen,</li> <li>• verstehen Studierende die historische Entwicklung der Zahlentheorie und haben somit einen Einblick in die Geschichte der Mathematik im Ganzen,</li> <li>• besitzen Studierende Grundkenntnisse für die modernen Anwendungen der Zahlentheorie, insbesondere im Bereich der Kryptographie</li> <li>• haben Studierende ihr Abstraktionsvermögen geschärft,</li> <li>• haben Studierende ihre Fähigkeit verbessert, komplexe Probleme in Teilprobleme zu zerlegen und diese zu lösen</li> </ul>					
<b>Inhalt:</b> Division mit Rest, Zahlendarstellung ganzer Zahlen, Euklidischer Algorithmus, eindeutige Primfaktorzerlegung, Modulare Arithmetik, RSA-Verschlüsselung, chinesischer Restsatz, Struktur der primen Restklassengruppe, pythagoräische Zahlentripel, 2-Quadrate-Satz von Fermat, Gaußsches Reziprozitätsgesetz, Pellische Gleichung, Kettenbrüche, euklidische Ringe, endliche Körper, Primzahltests					
<b>Lehrformen:</b> Vorlesung mit begleitenden Übungsgruppen					
<b>Prüfungsformen:</b> schriftliche Prüfung, 120-180min					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulklausur sowie erfolgreiches Referat / Thesenpapier / Vortrag etc. in Veranstaltung xy / Anwesenheit in mindestens xx von xx Terminen					

<b>Stellenwert der Note für die Endnote: 9 / 175</b>
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Dr. Viktoriya Ozornova</b>
<b>Sonstige Informationen:</b> <b>Literatur:</b> Müller-Stach, Piontkowski, "Elementare und algebraische Zahlentheorie", Vieweg+Teubner, 2011



<b>Einführung in die Kryptographie 1</b>					
<b>Modul-Nr:</b> 141022	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Fähigkeit zum abstrakten und logischen Denken.</p>					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden haben ein Verständnis über die wichtigsten symmetrischen Verschlüsselungsverfahren in der Praxis und Grundlagen der asymmetrischen Kryptographie. Darüberhinaus kennen sie die Denkweisen der modernen Kryptographie.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Selbständiges Lernen und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Mathematik</li> <li>• IT-Sicherheit</li> </ul>					
<p><b>Inhalt:</b> Es werden zunächst grundlegende Begriffe der Kryptographie und Datensicherheit eingeführt. Nach der Vorstellung einiger historischer Verschlüsselungsverfahren werden Stromchiffren behandelt. Den Hauptteil der Vorlesung bilden Blockchiffren und deren Anwendung. Als bedeutender Vertreter der symmetrischen Verfahren werden der Data Encryption Standard (DES) und der Advanced Encryption Standard (AES) behandelt. Gegen Ende der Vorlesung wird das Prinzip der asymmetrischen Kryptographie sowie das in der Praxis wichtigste asymmetrische Verfahren, der RSA-Algorithmus, vorgestellt.</p> <p>Neben den kryptographischen Algorithmen werden die notwendigen mathematischen Grundlagen (u.a. Ringe ganzer Zahlen, Euklidischer Algorithmus, endliche Körper) eingeführt.</p>					
<b>Lehrformen:</b>					

## KAPITEL 10. KRYPTOLOGIE

Vorlesung (+Übung)
<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Christof Paar</b> Prof. Dr.-Ing. Christof Paar
<b>Sonstige Informationen:</b> <b>Literatur:</b>  C. Paar, J. Pelzl: Understanding Cryptography: A Textbook for Students and Practitioners. Springer, 2009

<b>Einführung in die Kryptographie 2</b>					
<b>Modul-Nr:</b> 141022	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Inhalte der Vorlesung "Einführung in die Kryptographie 1"</p>					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden haben ein Verständnis der für die Praxis wichtigsten asymmetrischen Verschlüsselungsverfahren sowie den Einsatz von Krypto-Primitiven für die Realisierung von Sicherheitsdiensten.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Analytisches und logisches Denken</li> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Selbständiges Lernen und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Algorithmik</li> <li>• Mathematik</li> <li>• IT-Sicherheit</li> </ul>					
<p><b>Inhalt:</b> Einen wichtigen Teil der Vorlesung bilden asymmetrische kryptographische Verfahren basierend auf dem diskreten Logarithmusproblem. Es werden hier der Schlüsselaustausch nach Diffie-Hellman, die Elgamal-Verschlüsselung und Verfahren mit elliptischen Kurven behandelt. Nachfolgend werden Schemata und Protokolle basierend auf symmetrischen und asymmetrischen Primitiven entwickelt. Behandelt werden: Digitale Signaturen, Message Authentication Codes (MACs), Hash-Funktionen, Zertifikate, Protokolle zum Schlüsselaustausch sowie Sicherheitsdienste.</p>					
<p><b>Lehrformen:</b> Vorlesung (+Übung)</p>					

## KAPITEL 10. KRYPTOLOGIE

<b>Prüfungsformen:</b> Klausur
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Dr. Christof Paar</b> Prof. Dr. Gregor Leander, M. Sc. Marc Fyrbiak
<b>Sonstige Informationen:</b> Videomitschnitte unter <a href="http://www.crypto-textbook.com">www.crypto-textbook.com</a>  <b>Literatur:</b>  C. Paar, J. Pelzl: Understanding Cryptography: A Textbook for Students and Practitioners. Springer, 2009

<b>Kryptographie auf hardwarebasierten Plattformen</b>					
<b>Modul-Nr:</b> 141031	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung (+Übung)			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 90 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine</p> <p><b>Empfohlene Vorkenntnisse:</b>  Empfohlen: Grundlagen der Kryptographie und Datensicherheit, Basiswissen Digitaltechnik</p>					
<p><b>Lernziele (learning outcomes):</b> Die Studierenden kennen die Konzepte der praxisnahen Hardware- entwicklung mit abstrakten Hardwarebeschreibungssprachen (VHDL) und die Simulation von Hardwareschaltungen auf FPGAs. Sie beherrschen Stan- dardtechniken der hardwarenahen Prozessorentwicklung und sind zur Im- plementierung von symmetrischen und asymmetrischen Kryptosystemen auf modernen FPGA-Systemen in der Lage.</p> <p><b>Vermittelte Kompetenzen:</b> Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• IT-Sicherheit</li> <li>• Algorithmik</li> <li>• Mathematik</li> </ul>					
<p><b>Inhalt:</b> Kryptographische Systeme stellen aufgrund ihrer Komplexität ins- besondere an kleine Prozessoren und eingebettete Systeme hohe Anforde- rungen. In Kombination mit dem An- spruch von hohem Datendurchsatz bei geringsten Hardwarekosten ergeben sich hier für den Entwickler grundlegende Probleme, die in dieser Vorlesung beleuchtet werden sollen. Die Vor- lesung behandelt die interessantesten Aspekte, wie man aktuelle kryptographische Verfahren auf praxisnahen Hardwaresystemen implemen- tiert. Dabei werden Kryptosysteme wie die Blockchiffre AES, die Hashfunk- tionen SHA-1 sowie asymmetrische Systeme RSA und ECC behandelt. Wei- terhin werden auch spezielle Hardwareanforderungen wie beispielsweise der Erzeugung echten Zufalls (TRNG) sowie der Einsatz von Physically Unclo- nable Functions (PUF) besprochen. Die effiziente Implementierung dieser Kryptosysteme, insbesondere in Be- zug auf die Optimierung für Hochgeschwindigkeit, wird auf modernen FPGAs besprochen</p>					

## KAPITEL 10. KRYPTOLOGIE

und in praktischen Übungen mit Hilfe der Hardwarebeschreibungssprache VHDL umgesetzt. Vorlesungsbegleitend wird ein Moodle-Kurs angeboten, der zusätzliche Inhalte sowie die praktischen Übungen bereithält.

**Lehrformen:**

Vorlesung (+Übung)

**Prüfungsformen:**

Klausur

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote:** 5 / 175

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr.-Ing. Tim Güneysu**

**Sonstige Informationen:****Literatur:**

Keine

# Neuroinformatik

<b>Computersehen: Einführung</b>					
<b>Modul-Nr:</b> 310505 (Übung: 310515)	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> 4 oder 6 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vorlesung und Übungen Computersehen: Einführung			<b>Kontaktzeit:</b> 2 + 1 SWS	<b>Selbststudium:</b> 1 h	<b>Gruppen- größe:</b> 40 - 60 Studie- rende
<b>Teilnahmevoraussetzungen:</b> Keine					
<p><b>Lernziele (learning outcomes):</b> Ziel dieser Vorlesung ist es, die Grundlagen der digitalen Bildverarbeitung zu vermitteln. Die Studierenden sollen die Grundideen der verschiedenen Methoden verstehen und in die Lage versetzt werden, eigenständig Bildverarbeitungsalgorithmen in einer Programmiersprache ihrer Wahl umzusetzen. Vollständigkeit kann kein Ziel einer einführenden Vorlesung sein. Ein wesentliches Lernziel ist daher die Selbsterschließbarkeit moderner oder auch etablierter Bildverarbeitungsverfahren mit den Methoden, Konzepten und Herangehensweisen der Veranstaltung.</p>					
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"> <li>• Aufbau einer Kamera: Lochblendenmodell und moderne Kameras</li> <li>• Grundlegende Bildverarbeitungsoperationen: Punktoperatoren, Filter, Geometrische Transformationen</li> <li>• Punktdetektion und Punktdeskriptoren: SIFT, SURF, FAST</li> <li>• Bildregistrierung</li> <li>• Stereo-Sehen: Dynamische Programmierung und Semi-Global Matching</li> <li>• Einführung in das Maschinelle Lernen: Bildklassifikation</li> <li>• Echtzeitfähige Detektion: Viola-Jones-Detektor und Soft-Cascade</li> </ul>					

## KAPITEL 11. NEUROINFORMATIK

<b>Lehrformen:</b> 2 stündiges Vorlesungsformat mit zusätzlich wöchentlich zu bearbeitenden praktischen Übungen in Form von mathematischen Herleitungen, Verständnisfragen und Programmimplementierungen
<b>Prüfungsformen:</b> Klausur von 90 – 120 Minuten Länge mit mathematischen Herleitungen und Verständnisfragen
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur
<b>Stellenwert der Note für die Endnote:</b> 5 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Jun.-Prof. Sebastian Houben</b>



<b>Artificial Neural Networks</b>					
<b>Modul-Nr:</b> 310002 (Übung: 310012)	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> 3-6 Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 3 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b>  a) 150  b) 20
<p><b>Teilnahmevoraussetzungen:</b> calculus, linear algebra, statistics, programming</p> <p><b>Empfohlene Vorkenntnisse:</b></p> <p>Grundkenntnisse in der Infinitesimalrechnung, linearen Algebra, Statistik und Informatik. Erfahrung mit einer höheren Programmiersprache.</p>					
<p><b>Lernziele (learning outcomes):</b> 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</p> <ul style="list-style-type: none"> <li>• verstehen Studierende die theoretisch-mathematischen Grundlagen von KNN im Kontext des überwachten Lernens.</li> <li>• können Studierende selbstständig zwischen verschiedenen KNN unterscheiden und in einer Anwendungssituation das geeignete Verfahren auswählen.</li> <li>• können Studierende grundlegende Verfahren selbstständig in einer höheren Programmiersprache implementieren, sowie ihre eigene Implementierung und Standard-Implementierung anderer auf Daten anwenden.</li> <li>• können Studierende Ergebnis der KNN selbstständig interpretieren, insbesondere erkennen, wann sie unrealistisch sind.</li> </ul> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Selbständiges Lernen und Arbeiten</li> <li>• Analytisches und logisches Denken</li> <li>• Analyse und Interpretation von Daten</li> <li>• Kommunikation von quantitativen Zusammenhängen</li> </ul>					

## KAPITEL 11. NEUROINFORMATIK

Fachspezifische Kompetenzen:

- Verständnis der theoretisch-mathematischen Grundlagen von KNN im Kontext des überwachten Lernens.
- Unterscheidung zwischen verschiedenen KNN und Auswahl eines geeigneten Verfahrens in einer Anwendungssituation.
- Selbstständige Implementierung grundlegender Verfahren in einer höheren Programmiersprache, sowie Anwendung der eigenen und anderer Implementierung auf Daten.
- Selbstständige Analyse der Ergebnis der KNN interpretieren, insbesondere Erkennen, wann sie unrealistisch sind.

### **Inhalt:**

Verfahren: Struktur von Optimierungsproblemen, Regression, logistische Regression, biologische neuronal Netze, Modellselektion, universelle Approximationstheorem, Perzeptron, mehrschichtiges Perzeptron, Backpropagation, tiefe neuronale Netze, rekurrente neuronale Netze, Long-Short Term Memory, Hopfield Netze, Botzmann-Machine

Software: python, numpy, matplotlib, scikit-learn, tensorflow

### **Lehrformen:**

Vorlesung, Hausaufgaben, angeleitete Übungen am Computer

### **Prüfungsformen:**

Klausur

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

### **Verwendung des Moduls (in anderen Studiengängen):**

B.Sc. Elektrotechnik und Informationstechnik, M.Sc. Cognitive Science

### **Stellenwert der Note für die Endnote: 6 / 175**

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Sen Cheng**

### **Sonstige Informationen:**

#### **Literatur:**

- **Charu C. Aggarwal: “Neural Networks and Deep Learning”, Springer**
  - Aurélien Géron: “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques for Building Intelligent Systems”, O’Reilly
  - Skript

<b>Mathematics for Modeling and Data Analysis</b>					
<b>Modul-Nr:</b> 310503 (Übung: 310513)	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Mathematics for Modeling and Data Analysis			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b> 20
<p><b>Teilnahmevoraussetzungen:</b> Keine  <b>Empfohlene Vorkenntnisse:</b>    Basic knowledge of calculus and linear algebra.</p>					
<p><b>Lernziele (learning outcomes):</b>  After the successful completion of this course the students:</p> <ul style="list-style-type: none"> <li>• know the material covered in this course, see Content,</li> <li>• do have an intuitive understanding of the basic concepts and can work with that,</li> <li>• can communicate about all this in English.</li> </ul>					
<p><b>Inhalt:</b>  This course covers mathematical methods that are relevant for modeling and data analysis. Particular emphasis will be put on an intuitive understanding as is required for a creative command of mathematics. The following topics will be covered: Functions, vector spaces, matrices as transformations, systems of linear differential equations, and qualitative analysis of nonlinear differential equations, possibly also Bayesian theory and multiple integrals.</p>					
<p><b>Lehrformen:</b>  Lecture and tutorial</p>					
<p><b>Prüfungsformen:</b>  Oral exam, ca 20 min</p>					
<p><b>Voraussetzungen für die Vergabe von Kreditpunkten:</b>  Bestandene Modulabschlussklausur</p>					
<p><b>Stellenwert der Note für die Endnote:</b> 6 / 175</p>					
<p><b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>    <b>Prof. Dr. Laurenz Wiskott</b></p>					
<p><b>Sonstige Informationen:</b>  There is a lecture, which provides the content, and a tutorial, where you solve exercises and can deepen your understanding of the content. The exercises are solved in the tutorial in a group effort, not at home, which is the reason why it takes 3 hours rather than the usual 1.5 hours.</p>					



<b>Mathematical Psychology</b>					
<b>Modul-Nr:</b> 310506/16	<b>Credits:</b> 6 CP	<b>Workload:</b> 180 h	<b>Semester:</b> 4-6 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Übung			<b>Kontaktzeit:</b> 4 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppen- größe:</b> 10
<b>Teilnahmevoraussetzungen:</b> Basic knowledge of perception, decision making, learning and memory, e.g., “Cognition I + II”, “Learning”.					
<p><b>Lernziele (learning outcomes):</b> Perceptual, cognitive and motor processes can be captured and explained using simple mathematical or computer models. This class will introduce this powerful approach through a combination of interactive theory lectures and hands-on computer lab exercises. After successful completion of this course, students will be able</p> <ul style="list-style-type: none"> <li>• understand how to describe and model behavior quantitatively.</li> <li>• to do scientific programming.</li> <li>• design their own psychophysics experiments, as well as implement and analyze experiments using software.</li> </ul> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Independent learning and working</li> <li>• Analytical and logical thinking</li> <li>• Analysis und interpretation of data</li> <li>• Communication of quantitativ relationships</li> </ul> <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Modelling of behavioral experiments</li> <li>• Design, analysis und interpretation of behavioral experiments</li> <li>• Scientific programming</li> </ul>					
<p><b>Inhalt:</b> Models of perception, decision making, learning and memory. Quantitative methods: psychophysics, signal detection theory and neural network modeling.</p>					

## KAPITEL 11. NEUROINFORMATIK

<b>Lehrformen:</b> Lectures, guided computer exercises
<b>Prüfungsformen:</b> Course project.
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Adequate project report about course project.
<b>Verwendung des Moduls (in anderen Studiengängen):</b> B.Sc. Psychologie, M.Sc. Cognitive Science
<b>Stellenwert der Note für die Endnote:</b> 6 / 175
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  <b>Prof. Sen Cheng</b>

# Programmier- und Simulationstechnik

<b>Spieleentwicklung in C#</b>					
<b>Modul-Nr:</b> 310035	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Wintersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Praktikum			<b>Kontaktzeit:</b> 80h keine SWS	<b>Selbststudium:</b> 70 h	<b>Gruppen- größe:</b> max. 30 Teil- nehmer
<b>Teilnahmevoraussetzungen:</b> Keine					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> <li>• verstehen die Teilnehmer Grundlagen der objektorientierten Programmierung,</li> <li>• können die Teilnehmer praxisnahe Problemstellungen der Softwareentwicklung analysieren und fachgerecht lösen,</li> <li>• können die Teilnehmer ihre Ergebnisse in einem didaktisch gut aufbereiteten Vortrag ihren Kommilitonen vermitteln.</li> </ul>					
<b>Inhalt:</b> Einführung in die Programmiersprache C# Umsetzung des Erlernten durch Entwicklung einfacher Computerspiele in der Unity-Engine Erstellung eines selbstständig entworfenen Computerspiels in einem Gruppenprojekt					
<b>Lehrformen:</b> Vorlesung und begleitende Übung					
<b>Prüfungsformen:</b> Projektarbeit und Vortrag					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Erreichen von 75% der Übungspunkte, erfolgreiches Bearbeiten des Projekts mit Abschlussvortrag					
<b>Stellenwert der Note für die Endnote:</b> 5 / 175					

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Tobias Glasmachers**

Daniel Vonk



<b>Agent-based modeling in economics and business</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 5 CP	<b>Workload:</b> 150 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Jedes WS und SS 2020	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b>  a) Vorlesung  b) Seminar			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 120 h	<b>Gruppengröße:</b> 20 Teilnehmer, gemischt: Ang. Informatik und WiWi
<b>Teilnahmevoraussetzungen:</b> Gute Englischkenntnisse, Grundlegende Programmierkenntnisse					
<p><b>Lernziele (learning outcomes):</b> Das Modul verfolgt das Ziel, Studierende der angewandten Informatik und Studierende der Wirtschaftswissenschaft in die Methode der agenten-basierten Modellierung und Simulationstechniken einzuführen. Hierbei sollen Studierende relevante volkswirtschaftliche Fragestellungen mit quantitativen Ansätzen bearbeiten.</p> <p>Im Rahmen des Moduls erwerben Studierende folgende Kenntnisse:</p> <ol style="list-style-type: none"> <li>1) Überführung von wissenschaftlichen Fragestellungen aus dem Bereich Volkswirtschaftslehre in agenten-basierte Computersimulationen</li> <li>2) Praktische Arbeit mit agenten-basierten Modellen, Interpretation von Simulationsergebnissen und mögliche Limitationen des Ansatzes</li> <li>3) Grundlegendes Wissen, um eigene Modelle zu implementieren und Simulationen selbstständig durchzuführen</li> <li>4) Die Programmiersprache NetLogo</li> </ol>					
<p><b>Inhalt:</b> In der Vorlesung wird die Methode der agentenbasierten Computersimulation vorgestellt und gezeigt, wie sie zur Analyse komplexer ökonomischer Systeme angewendet werden kann. Die Methode der agenten-basierten Computersimulationen gewinnt in der Forschung sowie in der Praxis weiter an Relevanz. Dazu werden Beispiele aus der betriebswirtschaftlichen und der volkswirtschaftlichen Forschungsliteratur präsentiert. Während der Seminartermine werden verschiedene Modelle vorgestellt, die dann als Grundlage für Gruppendiskussionen und eigene Programmieraufgaben dienen. Hier steht die praktische Anwendung im Vordergrund.</p>					
<b>Lehrformen:</b> Vorlesung (+ Seminar)					
<b>Prüfungsformen:</b> Hausarbeit in Form eines Lab-Reports Vortrag im Seminarkolloquium					

**Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestehen der folgenden Leistungen:

- 1) Studienleistung (unbenotet)
- 2) Hausarbeit und Vortrag (benotet)

**Stellenwert der Note für die Endnote: 5 / 175**

**Modulbeauftragte/r und hauptamtlich Lehrende:**

**Roos**

**Sonstige Informationen:**

Verbindliche Anmeldung in der ersten Sitzung und durch Registrierung bei Moodle. Studierende der angewandten Informatik müssen sich beim Zentrum für ökonomische Bildung (ZfÖB) zum Modul anmelden, um eine Prüfungsleistung erbringen zu können.

**Literatur:**

Die Leseliste und geeignete Lehrbücher werden zu Beginn der Veranstaltung bekanntgeben.

# Vertiefungsseminar

<b>Seminar Computerlinguistik</b>					
<b>Modul-Nr:</b> 050025: Tools für den linguistischen Alltag: Automatisierung, Reproduzierbarkeit und Kollaboration; 050042: Dialogsysteme; 050042: Koreferenzauflösung	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vertiefungsseminare			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Linguistische Grundlagen, Linguistische Schwerpunkte und Linguistische Methoden					
<b>Lernziele (learning outcomes):</b> <ol style="list-style-type: none"> <li>1. Fortgeschrittene Kenntnisse in unterschiedlichen Bereichen der Computerlinguistik und Sprachtechnologie</li> <li>2. Fähigkeit, die für computerlinguistische Themen und Aufgabenstellungen jeweils geeigneten Analysemethoden sowie die entsprechenden Algorithmen und Implementationsverfahren auswählen, bewerten, und anwenden zu können</li> <li>3. generelle Kompetenz in eigenständiger computerlinguistischer Analyse, Modellierung und Implementierung sprachlicher Strukturen und Prozesse</li> </ol>					
<b>Vermittelte Kompetenzen:</b> Kernkompetenzen:					

## KAPITEL 13. VERTIEFUNGSSSEMINAR

- Analytisches und logisches Denken
- Fächerübergreifendes Denken und Arbeiten
- Literaturrecherche und Dokumentation
- Projekt- und Zeitmanagement
- Präsentation wissenschaftlicher Ergebnisse
- Selbständiges Lernen und Arbeiten

### **Inhalt:**

Gegenstand des Moduls sind unterschiedliche Themen aus den Bereichen Computerlinguistik und Sprachtechnologie, sprich deren spezifische Untersuchungsgegenstände und Methoden.

### **Seminare:**

Typische Themen einschlägiger und in unterschiedlichem Turnus angebotener Seminare sind u. a.

- Korpuslinguistik
- Data mining, Explorative Datenanalyse (mit R)
- Klassifikationsalgorithmen
- Natürlichsprachliche Systeme, Automatische Sprachverarbeitung, Automatische Lexikonakquise
- Automatische Textzusammenfassung, Textgenerierung
- Semantische Verarbeitung in der Computerlinguistik
- Morphologische und syntaktische Analyse in der Computerlinguistik

### **Lehrformen:**

Vertiefungsseminare

### **Prüfungsformen:**

Seminar-Arbeit

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

Bestandene Modulabschlussklausur

**Stellenwert der Note für die Endnote: 3 / 175**

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Johanna Marie Poppek, M. A.**

Roussel, Simonjetz, Klabunde

### **Sonstige Informationen:**

Die Teilnahme ist nur nach vorheriger Anmeldung in CampusOffice möglich. Diese Anmeldung muss im WS bis zum 01.10. und im SS bis zum 01.04. erfolgen. Zusätzlich ist die Anmeldung zur Prüfung über FlexNow innerhalb der gültigen Anmeldephase notwendig.

**Literatur:**

Wird im Kurs bekanntgegeben.

<b>Seminar Ingenieurinformatik</b>					
<b>Modul-Nr:</b> 125017	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 2-3 Sem.	<b>Turnus:</b> Winter- und Sommersemes- ter	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Seminar Ingenieurinformatik			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> beliebig
<p><b>Teilnahmevoraussetzungen:</b> Keine  <b>Empfohlene Vorkenntnisse:</b></p> <p>Mindestens ein Vertiefungsmodul der Ingenieurinformatik</p>					
<p><b>Lernziele (learning outcomes):</b>  Lernziel ist die selbständige Auseinandersetzung mit einem Thema aus dem Bereich der Ingenieurinformatik. Es soll die Fähigkeit der kritischen Auseinandersetzung mit einem Thema im Rahmen einer Fachdiskussion gefördert werden. Die Studierenden vertiefen sich in ein Gebiet, das ihnen zu Beginn des Seminars zugewiesen wird. Der Inhalt muß verständlich und fachlich korrekt wiedergegeben werden. Anschließende Diskussionen aller Seminarteilnehmer sollen dazu führen, dass sich die Studierenden kritisch mit dem fachlichen Inhalt des Vortrages auseinandersetzen.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> <li>• kennen Studierende die methodischen Ansätze und Techniken, um sich in eine Thematik gründlich einzuarbeiten. Dabei sollen die gewonnenen Erkenntnisse in verständlicher Form weitergegeben werden;</li> <li>• verstehen Studierende durch Bearbeitung einer Seminararbeit, wie die erarbeiteten Fakten, Informationen und Zusammenhänge in Rahmen einer Präsentation klar vermittelt werden können.</li> </ul> <p><b>Vermittelte Kompetenzen:</b>  Kernkompetenzen:</p> <ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Kritikfähigkeit</li> <li>• Literaturrecherche und Dokumentation</li> <li>• Projekt- und Zeitmanagement</li> <li>• Präsentation wissenschaftlicher Ergebnisse</li> </ul>					

## KAPITEL 13. VERTIEFUNGSSEMINAR

- Rhetorik und sprachliche Kompetenz
- Selbständiges Lernen und Arbeiten

### **Inhalt:**

Die im Rahmen eines Semesters angebotenen Seminarthemen werden zu Beginn des Semesters bekannt gegeben und decken Themen auf dem Gebiet der Ingenieurinformatik ab. Es wird darauf geachtet, dass die Themen einen engen Bezug zu aktuellen Problemstellungen, dem Stand der Technik und neuen Forschungserkenntnissen haben. Folgende Themenschwerpunkte werden behandelt

- Product Lifecycle Management
- Smart Product Engineering
- Product Development Methods
- Echtzeit Rendering
- Visual Analytics
- Simulationstechniken

### **Lehrformen:**

seminaristischer Unterricht; Erstellen einer Seminararbeit

### **Prüfungsformen:**

Präsentation der eigenen Seminararbeit

### **Voraussetzungen für die Vergabe von Kreditpunkten:**

- Erfolgreiche Bearbeitung einer Seminararbeit
- aktive Beteiligung mit Diskussionen an allen Präsentationen

**Stellenwert der Note für die Endnote: 3 / 175**

### **Modulbeauftragte/r und hauptamtlich Lehrende:**

**Prof. Dr. Markus König**  
Dr. K. Lehner

<b>Seminar Kryptologie und Theoretische Informatik</b>					
<b>Modul-Nr:</b> 150537: Se- minar zur Kryptographie	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> beliebiges Sem.	<b>Turnus:</b> Unregelmäßig	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Vertiefungsseminare			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> beliebig
<b>Teilnahmevoraussetzungen:</b> Keine					
<b>Empfohlene Vorkenntnisse:</b>  Kryptographie, Kryptanalyse, Algorithmik					
<b>Lernziele (learning outcomes):</b> Es werden aktuelle Forschungsarbeiten der Kryptologie und Algorithmik vorgestellt					
<b>Vermittelte Kompetenzen:</b> Kernkompetenzen:					
<ul style="list-style-type: none"> <li>• Fächerübergreifendes Denken und Arbeiten</li> <li>• Kritikfähigkeit</li> <li>• Literaturrecherche und Dokumentation</li> <li>• Projekt- und Zeitmanagement</li> <li>• Präsentation wissenschaftlicher Ergebnisse</li> <li>• Rhetorik und sprachliche Kompetenz</li> <li>• Selbständiges Lernen und Arbeiten</li> </ul>					
<b>Inhalt:</b> Inhalt entsprechend des gewählten Seminars.					
<b>Lehrformen:</b> Vertiefungsseminare					
<b>Prüfungsformen:</b> Seminar-Arbeit					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> Bestandene Modulabschlussklausur					
<b>Stellenwert der Note für die Endnote:</b> 3 / 175					
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>					



## KAPITEL 13. VERTIEFUNGSSSEMINAR

**Prof. Dr. Alexander May**

Prof. Dr. Alexander May, Prof. Dr. Hans Ulrich Simon, Prof. Dr. Eike Kiltz

**Sonstige Informationen:**

Kein Angebot im Winter 2019/2020

<b>Bachelor-Seminar Maschinelles Lernen</b>					
<b>Modul-Nr:</b> N/A	<b>Credits:</b> 3 CP	<b>Workload:</b> 90 h	<b>Semester:</b> 4 Sem.	<b>Turnus:</b> Sommersemester	<b>Dauer:</b> 1 Semester
<b>Lehrveranstaltungen:</b> Seminar (2 SWS)			<b>Kontaktzeit:</b> 2 SWS	<b>Selbststudium:</b> 60 h	<b>Gruppen- größe:</b> max. 20 Teil- nehmer
<b>Teilnahmevoraussetzungen:</b> Keine					
<b>Lernziele (learning outcomes):</b> Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> <li>• kennen die Teilnehmer Algorithmen und Anwendungen des maschinellen Lernens</li> <li>• können die Teilnehmer Lehrinhalte und Forschungsergebnisse eigenständig in einem didaktisch wohl aufbereiteten Vortrag vermitteln,</li> <li>• können die Teilnehmer konstruktives Feedback formulieren und entgegennehmen.</li> </ul>					
<b>Inhalt:</b> Das Seminar vertieft diverse Inhalt aus dem breiten Bereich des maschinellen Lernens. Weiterhin werden guter Vortragsstil und das üben sowie die Entgegennahme konstruktiver Kritik geübt.					
<b>Lehrformen:</b> Seminarvorträge und Feedbackrunden					
<b>Prüfungsformen:</b> Seminarvortrag					
<b>Voraussetzungen für die Vergabe von Kreditpunkten:</b> erfolgreiche Präsentation des Seminarvortrags					
<b>Stellenwert der Note für die Endnote:</b> 3 / 175					
<b>Modulbeauftragte/r und hauptamtlich Lehrende:</b>  Prof. Dr. Tobias Glasmachers					