

**Master Studiengang
Angewandte Informatik**

PO 13

Modulhandbuch

Erstellt am: 6. Dezember 2019

Wo finde ich Hilfe während des Studiums?

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 Masterstudiums ist die Vermittlung von Kenntnissen auf dem Gebiet der Angewandten Informatik, um komplexe Ingenieur Tätigkeiten selbstständig und verantwortlich durchführen zu können. Der Masterstudiengang führt damit zu einer Berufsqualifizierung, die für eine Mitarbeit in Forschung und Entwicklung mit Führungsverantwortung nötig ist. Er vermittelt zudem die notwendigen Kenntnisse für wissenschaftliche Arbeiten auf Promotionsniveau. Durch die Masterprüfung soll festgestellt werden, ob der Kandidat bzw. die Kandidatin fundierte Kenntnisse und die Fähigkeit zur selbstständigen Anwendung anspruchsvoller wissenschaftlicher Methoden erlernt hat. Die Studierenden sollen zur kritischen Einordnung der wissenschaftlichen Erkenntnisse sowie zu verantwortlichem, interdisziplinärem Denken und Handeln befähigt werden und sollen komplexe Probleme der Angewandten Informatik analysieren und Lösungen erarbeiten können. Erweiterte Sprachkenntnisse und Studienaufenthalte im Ausland sind erwünscht.

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,?). Im Masterstudium gibt es keine Pflichtmodule. Die im Wahlpflichtbereich angesiedelten Module erweitern das Grundlagenwissen aus dem Bachelorstudium, um den spezialisierteren Vertiefungsmodulen folgen zu können. Insgesamt bietet das Studium zahlreiche Wahlmöglichkeiten für die Studierenden, um ein individuelles Studienprofil zu entwickeln.

Prüfungsformen

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

Studienplan Master Angewandte Informatik PO 13

Nr	Modul	Lehrveranstaltung	Mind. Umfang Modul (LP)	Semester der Modulprüfung	Autom. Anmeldung im 3. FS	Autom. Wiederanmeldung	Bewertung
Wahlpflichtbereich							
1	Wahlpflichtmodule	Fächer aus dem Wahlpflichtkatalog *	20	1-3	Ja	Ja	benotet
Vertiefungsbereich							
2	Vertiefungsmodule	Fächer aus der Vertiefung **	45	1-3	Nein	Ja	benotet
3	Seminar	Masterseminar ***	3	1-3	Nein	Nein	benotet
4	Freie Wahlfächer	frei wählbar ****	8	1-3	Nein	Nein	unbenotet
			≥4	aus den obigen Bereichen			
Studienprojekt							
5	Master-Studienprojekt	Master-Studienprojekt	10	1-3	Nein	Nein	benotet
Masterarbeit							
6	Masterarbeit und Kolloquium	Masterarbeit und Kolloquium	30	4	Nein	Nein	benotet
Summe:			120				

- * Die Liste der wählbaren Wahlpflichtmodule befindet sich im jeweils aktuellen Modulhandbuch, welches vom Gemeinsam beschließenden Ausschuss verabschiedet wird.
Es müssen Module im Umfang von mindestens 20 LP gewählt werden.
- ** Die Liste der wählbaren Vertiefungsmodule befindet sich im jeweils aktuellen Modulhandbuch, welches vom Gemeinsam beschließenden Ausschuss verabschiedet wird.
Es müssen Module im Umfang von mindestens 45 LP gewählt werden.
- *** Die Liste der wählbaren Masterseminare befindet sich im jeweils aktuellen Modulhandbuch, welches vom Gemeinsam beschließenden Ausschuss verabschiedet wird.
Es müssen Seminare im Umfang von mindestens 3 LP gewählt werden.
- **** Hier können nahezu alle Veranstaltungen des Vorlesungsverzeichnisses der RUB, sowie Veranstaltungen im Rahmen der UAMR gewählt werden.
Es müssen Veranstaltungen im Umfang von mindestens 8 LP gewählt werden. Aus der Fakultät Wirtschaftswissenschaften sind nicht alle Veranstaltungen für die AI geöffnet. Eine entsprechende Auflistung befindet sich im aktuellen Modulhandbuch.

Master Angewandte Informatik PO13

Lehrveranstaltung	Einheit	Umfang Modul (LP)	Semester	Autom. Anmeldung	Autom. Wiederanmeldung	Bewertung
Wahlpflichtmodule						
Effiziente Algorithmen	Mathe	9	SS	Nein	Ja	benotet
Groupware und Wissensmanagement (nicht im WS 19/20!)	AW	5	WS	Nein	Ja	benotet
Komplexitätstheorie	Mathe	9	WS	Nein	Ja	benotet
Parallel Computing	Baulng	6	SS	Nein	Ja	benotet
Nebenläufige Programmierung	extern	5	SS	Nein	Ja	benotet
Machine Learning: Supervised Methods	INI	6	SS	Nein	Ja	benotet
Machine Learning: Unsupervised Methods	INI	6	WS	Nein	Ja	benotet
Seminare						
Ingenieurinformatik	MB/Baulng	3	WS/SS	Nein	Nein	benotet
Kryptologie und Theoretische Informatik (Angebot siehe Modulhandbuch)	Mathe	3	unregelmäßig	Nein	Nein	benotet
Seminar Algorithmen	Mathe	5	SS	Nein	Nein	benotet
Computerlinguistik (Angebot siehe Modulhandbuch)	Philologie	3	WS/SS	Nein	Nein	benotet
Operations Research (Management Science) letztmalig im SS 20!	WiWi	5	WS/SS	Nein	Nein	benotet
Computational Cognitive Modeling (nicht im WS 19/20!)	INI	3	WS	Nein	Nein	benotet
Bioinformatik	PURE/ProDi	3	WS	Nein	Nein	benotet
Vertiefungsmodule						
Ingenieurinformatik						
Design Optimization	Baulng	6	WS	Nein	Ja	benotet
Design soziotechnischer Informationssysteme	AW	5	SS	Nein	Ja	benotet
IT im Engineering	MB	6	WS	Nein	Ja	benotet
Künstliche Intelligenz für Ingenieure	ETIT	5	SS	Nein	Ja	benotet
Product Lifecycle Management	MB	6	SS	Nein	Ja	benotet
Grundlagen der Verkehrsplanung und Verkehrstechnik	Baulng	5	WS	Nein	Ja	benotet
Wissensbasierte Methoden	Baulng	6	SS	Nein	Ja	benotet
Grundlagen der automatischen Spracherkennung	ETIT	6	SS	Nein	Ja	benotet
Programmier- und Simulationstechnik						
Fundamentals of GPU Programming	ETIT	5	WS	Nein	Ja	benotet
Simulationstechnik	Baulng	5	WS	Nein	Ja	benotet
3D-Simulation in der Automatisierungstechnik	MB	6	WS	Nein	Ja	benotet
Complexity Economics and Agent-based Modeling	WiWi	10	WS	Nein	Ja	benotet
Neuroinformatik						
Autonomous Robotics: Action, Perception and Cognition	INI	6	SS	Nein	Ja	benotet
Computational Neuroscience: Neural Dynamics	INI	6	WS	Nein	Ja	benotet
Computational Neuroscience: Vision and Memory	INI	6	SS	Nein	Ja	benotet
Machine Learning: Evolutionary Algorithms	INI	6	WS	Nein	Ja	benotet
Master-Praktikum: Autonomous Robotics	INI	3	WS/SS	Nein	Ja	benotet
Computer Vision: Deep Learning Lab Course	INI	2	SS	Nein	Ja	benotet
Computer Vision: Deep Learning	INI	6	WS	Nein	Ja	benotet
Kryptologie und Theoretische Informatik						
Kryptanalyse 1	Mathe	5	SS	Nein	Ja	benotet
Kryptanalyse 2 (nicht im WS 19/20!)	Mathe	4	WS	Nein	Ja	benotet
Algorithmische Geometrie	Mathe	6	WS	Nein	Ja	benotet
Kryptographie	Mathe	9	WS	Nein	Ja	benotet
Kryptographische Protokolle	Mathe	5	SS	Nein	Ja	benotet
Quantenalgorithmen (nicht im WS 19/20!)	Mathe	6	WS	Nein	Ja	benotet
Theorie des maschinellen Lernens	Mathe	9	SS	Nein	Ja	benotet
Betriebssystemsicherheit	ETIT	5	WS	Nein	Ja	benotet
Operations Research & Management						
Rationales Entscheiden (letztmalig im WS 19/20!)	WiWi	5	WS	Nein	Ja	benotet
Industrielles Kundenmanagement	MB	5	SS	Nein	Ja	benotet

Bioinformatik							
Bioinformatik in der Proteomik I	MPC	5	WS	Nein	Ja	benotet	
Bioinformatik in der Proteomik II	MPC	5	SS	Nein	Ja	benotet	
Bioimage Informatics	Biologie	5	WS	Nein	Ja	benotet	
Master-Praktikum: Big Data in der Bioinformatik	MPC	5	WS/SS	Nein	Ja	benotet	
Algorithmische Bioinformatik	TUDo	6	WS	Nein	Ja	benotet	
Computational Omics	TUDo	6	SS	Nein	Ja	benotet	
Algorithmen auf Sequenzen	TUDo	5	SS	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
 MPC: Medizinisches Proteom Center
 WIWI: Fakultät für Wirtschaftswissenschaft

SS: Sommersemester
 WS: Wintersemester

LP: Leistungspunkte (auch Creditpoints oder CP)

Inhaltsverzeichnis

1	Wahlpflichtmodule	1
1.1	Effiziente Algorithmen	1
1.2	Groupware und Wissensmanagement	3
1.3	Komplexitätstheorie	5
1.4	Parallel Computing	7
1.5	Nebenläufige Programmierung	9
1.6	Machine Learning: Supervised Methods	12
1.7	Machine Learning: Unsupervised Methods	14
2	Masterarbeit und Kolloquium	16
2.1	Masterarbeit	16
3	Ingenieurinformatik	18
3.1	Design Optimization	18
3.2	Design sozio-technischer Informationssysteme	20
3.3	IT im Engineering	22
3.4	Künstliche Intelligenz für Ingenieure	24
3.5	Product Lifecycle Management	26
3.6	Grundlagen der Verkehrsplanung und Verkehrstechnik	28
3.7	Wissensbasierte Methoden	30
3.8	Grundlagen der automatischen Spracherkennung	32
4	Programmier- und Simulationstechnik	34
4.1	Fundamentals of GPU Programming	34
4.2	Simulationstechnik	36
4.3	3D-Simulation in der Automatisierungstechnik	38
4.4	Complexity Economics and Agent-based Modeling	40
5	Neuroinformatik	42
5.1	Autonomous Robotics: Action, Perception and Cognition	42
5.2	Computational Neuroscience: Neural Dynamics	44
5.3	Computational Neuroscience: Vision and Memory	46
5.4	Machine Learning: Evolutionary Algorithms	48
5.5	Master-Praktikum: Autonomous Robotics	50
5.6	Computer Vision: Deep Learning Lab Course	52
5.7	Computer Vision: Deep Learning	54

6	Kryptologie und Theoretische Informatik	56
6.1	Kryptanalyse 1 (Einführung in die asymmetrische Kryptanalyse)	56
6.2	Kryptanalyse 2 (Asymmetrische Kryptanalyse)	58
6.3	Algorithmische Geometrie	60
6.4	Kryptographie	62
6.5	Kryptographische Protokolle	64
6.6	Theorie des maschinellen Lernens	66
6.7	Betriebssystemsicherheit	68
7	Operations Research	70
7.1	Industrielles Kundenmanagement	70
7.2	Rationales Entscheiden	73
8	Bioinformatik	75
8.1	Bioinformatik in der Proteomik I	75
8.2	Bioinformatik in der Proteomik II	78
8.3	Bioimage Informatics	81
8.4	Master-Praktikum Big Data in der Bioinformatik	83
8.5	Algorithmische Bioinformatik	85
8.6	Computational Omics	87
8.7	Algorithmen auf Sequenzen	90
9	Masterseminare	92
9.1	Seminar Ingenieurinformatik	92
9.2	Seminar Kryptologie und Theoretische Informatik	94
9.3	Seminar Computerlinguistik	96
9.4	Seminar Operations Research (Management Science)	98
9.5	Seminar Computational Cognitive Modeling	100
9.6	Seminar Bioinformatik	102
9.7	Seminar Algorithmen	104
10	Studienprojekt	105
10.1	Master-Studienprojekt	105
11	Freie Wahlfächer	107
11.1	Freie Wahlfächer	107

Wahlpflichtmodule

Effiziente Algorithmen					
Modul-Nr: 150320 (Übung: 150321)	Credits: 9 CP	Workload: 270 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: V: 4 SWS; Ü: 2 SWS	Selbststudium: 180 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Die Inhalte der Veranstaltung "Datenstrukturen".					
Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls: Die Studierende					
<ul style="list-style-type: none"> • Kennen, auswählen und nutzen grundlegende Datenstrukturen und Graphenalgorithmen • Sind in der Lage Analysetechniken (Korrektheitsbeweise und Laufzeitanalyse) zu erläutern und beurteilen • Können auch bei praktischen Problemen entscheiden, welche der vermittelten Methoden/Algorithmen/Datenstrukturen anwendbar sind und diese nach Effizienz (insb. Laufzeit der Algorithmen) bewerten • Können konkrete Anwendungsprobleme modellieren und bei Bedarf diese Algorithmen weiter entwickeln 					
Vermittelte Kompetenzen:					
Kernkompetenzen:					
<ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten 					

KAPITEL 1. WAHLPFLICHTMODULE

Fachspezifische Kompetenzen:

- Algorithmik
- Graphentheorie
- Mathematik
- Optimierungsmethoden

Inhalt:

Die Lehrveranstaltung kann sowohl in das Gebiet der praktischen als auch in das Gebiet der theoretischen Informatik eingeordnet werden. Die zentralen Themen sind die Folgenden:

- Berechnung kürzester Pfade in Digraphen
- Berechnung eines maximalen Flusses in einem Transportnetzwerk
- Berechnung einer optimalen Lösung bei einem Zuordnungsproblem (auch Matching-Problem genannt)

Darüberhinaus beschäftigen wir uns mit Anwendungen dieser grundlegenden Probleme.

Lehrformen:

Vortrag der Lehrenden in der Vorlesung, Gruppenarbeit in den Übungen

Prüfungsformen:

Klausur (120 Minuten)

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 9 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

PD Dr. Daniela Kacso

Groupware und Wissensmanagement					
Modul-Nr: 260080	Credits: 5 CP	Workload: 150 h	Semester: 1-3 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung und Übung integriert			Kontaktzeit: 4 SWS	Selbststudium: 105 h	Gruppen- größe: 4-8
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse: Mensch-Maschine-Interaktion</p>					
<p>Lernziele (learning outcomes): Insbesondere durch die Nutzung neuer Technologien sollen die Studierenden erfahren, wie „Groupware und Wissensmanagement“ in der Praxis funktioniert und Konzepte erarbeiten, wie die Technik, aber auch die dazugehörigen sozio-technischen Prozesse gestaltet werden müssen, um für die Nutzer/innen nutzbar und nützlich zu sein. Dieser Einsatz ändert sich jeweils im Vergleich zum Vorjahr, da er von der Art der Projektarbeit abhängt.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Projekt- und Zeitmanagement • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Entwicklungsprozesse 					
<p>Inhalt: In zunehmendem Maße kommen Computersysteme nicht als Individualplatzlösung, sondern in Netzwerken zum Einsatz. Diese unterstützen gemeinsames Arbeiten und Lernen und werden unter den Schlagworten Groupware und Wissensmanagement zusammengefasst. Bei der Gestaltung und Einführung solcher Systeme wird man sich immer auch mit Fragen der Organisations- und Kommunikationsveränderungen befassen müssen. Die Vorlesung ist in vier thematische Blöcke unterteilt, die aufeinander aufbauen:</p> <ul style="list-style-type: none"> • Interdisziplinäre Grundlagen zum Verständnis von menschlicher Kommunikation, Kooperation und Organisation • Groupwaresysteme sowie Methoden und Werkzeuge • Mechanismen zur Kommunikationsunterstützung sowie zur Kooperation und Koordination 					

KAPITEL 1. WAHLPFLICHTMODULE

- Social Software/Web 2.0,
- Computer supported collaborative Learning
- Wissensmanagement in Organisationen

Lehrformen:

Abwechslung von Vorlesungsinhalten, Seminarbeiträgen der Studierenden, Präsenzübungen und Erörterung der Projektbearbeitung

Prüfungsformen:

Seminarbeitrag, Präsentation der Bearbeitung einer Projektaufgabe in Kleingruppen, Prüfungsgespräch zu Überprüfung der Eigenleistung bei der Projektarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Ein Seminarbeitrag (25% der Endnote) und Präsentation der Projektarbeit (75%)

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Thomas Herrmann
Markus Jelonek Msc.

Sonstige Informationen:**Literatur:**

1. Schwabe, G., Streitz, N., Unland, R. (Hrsg.) (2001): CSCW Kompendium. Lehr- und Handbuch zum computergestützten kooperativen Arbeiten. Heidelberg u.a.: Springer. S. 25 – 35.
2. Haake, Jörg; Schwabe, Gerd; Wessner, Martin (Hrsg.) (2012): CSCL-Kompendium 2.0: Lehr- und Handbuch zum computerunterstützten kooperativen Lernen. München: Oldenbourg Verlag.

Komplexitätstheorie					
Modul-Nr: 150262 (Übung: 150263)	Credits: 9 CP	Workload: 270 h	Semester: 1 oder 3 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung Komplexitätstheorie b) Uebungen zur Vorlesung			Kontaktzeit: 6 SWS	Selbststudium: 180 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse: Vorlesung über Theoretische Informatik					
Lernziele (learning outcomes): In der Vorlesung ergeben sich fundamentale Einsichten zur Komplexität von Berechnungsproblemen. Die Studierenden erwerben die Fähigkeit zwischen effizient berechenbaren und inhärent schweren Problemen zu unterscheiden. Zudem erlernen sie Techniken zur Einordnung eines Problems in die zugehörige Komplexitätsklasse. Nach dem erfolgreichen Abschluss des Moduls					
<ul style="list-style-type: none"> • kennen Studierende die wichtigsten Zeit- und Platzkomplexitätsklassen sowohl für deterministische als auch nichtdeterministische bzw. probabilistische Maschinenmodelle, • sind Studierende mit der Technik der effizienten Problemreduktionen vertraut, • verstehen Studierende komplexitätstheoretische Studien von Berechnungsproblemen und können im Einzelfall solche Studien selber erstellen. 					
Inhalt: Die Komplexitätstheorie stellt sich die Aufgabe Berechnungsprobleme anhand des zu ihrer Lösung erforderlichen Verbrauchs an Rechenzeit oder Speicherplatz in Klassen einzuordnen. Probleme von (annähernd) gleicher Komplexität landen dabei in derselben Klasse. Gegenstand der Vorlesung sind hauptsächlich die Komplexitätsklassen zwischen P und PSpace wie zum Beispiel die Klasse NP. Hierbei bezeichnet P die Klasse der in Polynomialzeit und PSpace die Klasse der mit polynomiell beschränktem Speicherplatz erkennbaren Sprachen. NP ist das nichtdeterministische Pendant zu P und bezeichnet die Klasse der nichtdeterministisch in Polynomialzeit erkennbaren Sprachen. Die Klasse enthält eine Vielzahl von grundlegenden Problemen aus verschiedenen Wissenschaftsbereichen. Eine der wichtigsten ungeklärten Fragen der theoretischen Informatik ist, ob die Klassen P und NP überhaupt verschieden sind. In der Vorlesung behandeln wir eingehend die NP-Vollständigkeitstheorie, die sich mit schwersten Problemen innerhalb NP beschäftigt. Weitere Themen sind die polynomielle Hierarchie von Stockmeyer, schwerste Probleme in PSpace und schließlich randomisierte Algorithmen bzw. Approximationsalgorithmen und die jeweils dazu passenden Komplexitätsklassen.					

KAPITEL 1. WAHLPFLICHTMODULE

Lehrformen: Vorlesung mit begleitenden Übungen
Prüfungsformen: mündliche Prüfung mit einer Dauer von 25-30 Minuten
Voraussetzungen für die Vergabe von Kreditpunkten: bestandene mündliche Prüfung
Verwendung des Moduls (in anderen Studiengängen): Studiengang Mathematik
Stellenwert der Note für die Endnote: 9 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Hans Ulrich Simon
Sonstige Informationen: Literatur: Skriptum zur Vorlesung

Parallel Computing					
Modul-Nr: 127501	Credits: 6 CP	Workload: 180 h	Semester: 2 / 4 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Parallel Computing			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Programmierkenntnisse in den Sprachen C, C++ oder Java</p>					
<p>Lernziele (learning outcomes):</p> <p>Goals of the course “Parallel Computing” include a basic understanding of the theoretical foundation for efficient parallel algorithms, the architecture of parallel hardware systems and current parallel software paradigms used in research and industry today. Also, students learn how to develop parallel algorithms and implement them using state-of-the-art software systems and programming tools. In particular, numerically intensive engineering applications are a focus within the lecture itself as well as a target for the team projects carried out by the students of this course. Further, students must clearly present their projects results in a classroom setting to an audience with various technical background (the course is attended by students from applied computer science as well as by students from computational engineering).</p> <p>When the students have successfully complete this course,</p> <ul style="list-style-type: none"> • students will be familiar with the development of parallel algorithms and parallel software systems and have a good understanding of how to realize efficient parallel programs; • students will have a good understanding of the basics of parallel programming to be able to, for example, select a proper parallel software system or tool that is appropriate for a given numerical engineering task, or be able to discuss possible solutions concerning necessary hardware or software requirements at a competent level. 					
<p>Inhalt:</p> <ul style="list-style-type: none"> • Introduction to parallel computing using examples from the field of engineering • Theoretical foundations of parallel computing (concurrency, parallel processes, deadlocks, Amdahl’s Law, Flynn’s taxonomy, efficiency metrics, memory models, etc.) • Parallel programming based on “shared memory” using the OpenMP application programming interface • Parallel programming based on “distributed memory” based the Message Passing Interface (MPI) 					

KAPITEL 1. WAHLPFLICHTMODULE

- Hardware based parallel programming based on the General Purpose Graphical Processing Unit (GPGPU)
- Application of parallel programming paradigms to solve engineering tasks as a team project

Lehrformen:

- lectures
- exercises using computers
- team projects

Prüfungsformen:

- completion of a team project (2-4 students)
- presentation of project results
- oral examination

Voraussetzungen für die Vergabe von Kreditpunkten:

- successful completion of a team project
- presentation of project results in the classroom
- oral examination

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. M. König, Dr. K. Lehner

Sonstige Informationen:

Literatur:

1. A. Schill, Th. Springer, "Verteilte Systeme", Springer-Verlag, 2007
2. Th. Raubner, G. Rüniger, "Parallele Programmierung", Springer-Verlag, 2007
3. G. Bengel, et. al., "Masterkurs Parallele und Verteilte Systeme", Vieweg+Teubner, 2008

Nebenläufige Programmierung					
Modul-Nr: 310509	Credits: 5 CP	Workload: 150 h	Semester: 1-3 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung: 2 SWS b) Übung: 1 SWS			Kontaktzeit: 3 SWS	Selbststudium: 120 h	Gruppen- größe: 30 Studierende
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Informatik 1, Informatik 2, Web-Engineering, Software-Engineering</p>					
<p>Lernziele (learning outcomes):</p> <p>Die Veranstaltung vermittelt grundlegende Fähigkeiten und Techniken, um nebenläufige Programme sicher entwickeln zu können. Es werden softwaretechnische Entwurfsmuster behandelt und vertieft, welche bekannte Probleme bei nebenläufigen Programmen wie zum Beispiel die Verklemmung vermeiden lassen. Der Zuhörer sollte am Ende der Veranstaltung nach einem erfolgreichen Abschluss der Klausur unter anderem in der Lage sein:</p> <ul style="list-style-type: none"> • Die Performanz von Programmen durch den Einsatz der nebenläufigen Programmierung zu verbessern • Bestehende Programme zu analysieren und mögliche Fehler zu erkennen • Die Sprachmerkmale und Schnittstellen von JAVA für die nebenläufige Programmierung sicher anzuwenden <p>Fach- und Methodenkompetenz im Detail. Die Studierenden sind in der Lage,</p> <ul style="list-style-type: none"> • die Begriffe Anwendung, Programm, Modul, Prozess und Thread voneinander zu unterscheiden (Wissen/Verstehen), • zu verstehen, warum Threads in manchen Situationen synchronisiert werden müssen (Wissen/Verstehen), • Synchronisation als Ursache von Verklemmungen zu begreifen (Wissen/Verstehen), • Verklemmungen durch Anwendung geeigneter Entwurfsmuster zu vermeiden (Erschaffen/Kreieren), • wichtige Klassen des Java APIs in eigenen nebenläufigen Programmen nutzen können (Anwenden), • die UML-Notation zur Modellierung nebenläufiger Sachverhalte zu beherrschen (Wissen/Verstehen sowie Erschaffen/Kreieren) 					

Vermittelte Kompetenzen:

Fachspezifische Kompetenzen:

- Algorithmik
- Software-Entwurfsmethoden
- Programmieren

Technologien:

- CUDA
- Java
- UML

Inhalt:

Moderne Hardware-Architekturen lassen sich nur durch den Einsatz nebenläufiger Programme richtig ausnutzen. Die nebenläufige Programmierung garantiert bei richtiger Anwendung eine optimale Auslastung der Hardware. Jedoch sind mit einem sorglosen Einsatz dieser Technik auch viele Risiken verbunden. Die Veranstaltung stellt Vorteile und Probleme nebenläufiger Programme dar und zeigt, wie sich die Performanz von Programmen verbessern lässt:

- Nebenläufigkeit: Schnelleinstieg
 - Anwendungen vs. Prozesse
 - Programme und ihre Ausführung
 - Vorteile und Probleme von nebenläufigen Programmen
 - * Verbesserung der Performanz
 - * Synchronisation
 - * Realisierung kritischer Abschnitte
 - * Monitore
 - * Lebendigkeit
 - * Verklemmungen
- Threads in Java
- UML-Modellierung von Nebenläufigkeit
- Neues zur Nebenläufigkeit in Java 5 und Java 6
- Realisierung von Nebenläufigkeit
- Fortgeschrittene Java-Konzepte für Nebenläufigkeit
- Synchronisationsobjekte

KAPITEL 1. WAHLPFLICHTMODULE

<ul style="list-style-type: none">• Datenstrukturen und Algorithmen für nebenläufige Programme• Entwurfsmuster
Lehrformen: Vorlesung mit Medieneinsatz (Powerpoint/Projektion) und geringem Tafelanschrieb
Prüfungsformen: Schriftliche Klausurarbeit (90min)
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Verwendung des Moduls (in anderen Studiengängen): Masterstudiengang IT-Sicherheit/Informationstechnik
Stellenwert der Note für die Endnote: 4.17%
Modulbeauftragte/r und hauptamtlich Lehrende: Studiendekan AI Dr. Doga Arinir
Sonstige Informationen: Das Modul setzt grundlegende Kompetenzen im Bereich der Programmierung voraus. Kenntnisse in der Java-,C++- oder C#-Programmierung sind empfehlenswert, werden aber nicht vorausgesetzt.

Machine Learning: Supervised Methods					
Modul-Nr: 310508 (Übung: 310518)	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: flipped classroom			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig, zuletzt 80 Teilnehmer
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>Vorlesung: Mathematics for Modeling and Data Analysis</p>					
<p>Lernziele (learning outcomes): Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • verstehen die Teilnehmer die Grundlagen der statistischen Lerntheorie, • kennen die Teilnehmer die wichtigsten Algorithmen des überwachten statistischen Lernens und können diese auf Lernprobleme anwenden, • kennen die Teilnehmer Stärken und Beschränkungen verschiedenen Lernverfahren, • können die Teilnehmer Standardsoftware zum maschinellen Lernen zur Lösung neuer Probleme einsetzen. 					
<p>Inhalt: Grundlagen der statistischen Lerntheorie, Querschnitt der wichtigsten Algorithmen des maschinellen Lernens, konkrete Problemlösung mit Standardsoftware</p>					
<p>Lehrformen: flipped classroom</p>					
<p>Prüfungsformen: Klausur 90 Minuten, Wiederholungsklausur nach 6 Monaten</p>					
<p>Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur</p>					
<p>Verwendung des Moduls (in anderen Studiengängen): Master computational engineering Master Medizinphysik Master Physik Master Biologie Sämtliche Ingenieurstudiengänge Die Veranstaltung wird auch regelmäßig von Studierenden der Chemie, der Mathematik, aus cognitive science, von Doktoranden, sowie von Studierenden der TU Dortmund (insbesondere automation and robotics) nachgefragt.</p>					

KAPITEL 1. WAHLPFLICHTMODULE

Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Jun.-Prof. Dr. Tobias Glasmachers
Sonstige Informationen: Es bietet sich an, diese Vorlesung in Kombination mit der Veranstaltung "Machine Learning: Unsupervised Methods", angeboten jeweils im Wintersemester, zu hören.

Machine Learning: Unsupervised Methods					
Modul-Nr: 310003 (Übung: 310013)	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Machine Learning: Unsupervised Methods			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: 30
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>The mathematical level of the course is mixed but generally high. The tutorial is almost entirely mathematical. Mathematics required include calculus (functions, derivatives, integrals, differential equations, ...), linear algebra (vectors, matrices, inner product, orthogonal vectors, basis systems, ...), and a bit of probability theory (probabilities, probability densities, Bayes' theorem, ...).</p>					
<p>Lernziele (learning outcomes): After the successful completion of this course the students:</p> <ul style="list-style-type: none"> • know a number of important unsupervised learning methods, • can discuss and decide which of the methods are appropriate for a given data set, • understand the mathematics of these methods, • can communicate about all this in English. <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Mathematik • Künstliche Intelligenz und Maschinelles Lernen 					
<p>Inhalt: This course covers a variety of unsupervised methods from machine learning such as principal component analysis, independent component analysis, vector quantization, clustering, self-organizing maps, growing neural gas, Bayesian theory and graphical models. We will also briefly discuss reinforcement learning.</p>					

KAPITEL 1. WAHLPFLICHTMODULE

Lehrformen: Flipped/inverted classroom
Prüfungsformen: Oral exam, ca 20 min
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Laurenz Wiskott
Sonstige Informationen: 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.

Masterarbeit und Kolloquium

Masterarbeit					
Modul-Nr: N/A	Credits: 30 CP	Workload: 900 h	Semester: beliebiges Sem.	Turnus: immer	Dauer: 1 Semester
Lehrveranstaltungen: Praktikum/Projektarbeit			Kontaktzeit: 0 SWS	Selbststudium: 900 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: 80 LP und abgeschlossener Wahlpflichtbereich, siehe Prüfungsordnung</p> <p>Empfohlene Vorkenntnisse: Fachspezifisch von der Themenstellung abhängig.</p>					
<p>Lernziele (learning outcomes): Die Masterarbeit soll problemorientiert ausgerichtet sein und damit innovative Lösungsansätze bzw. Lösungen beinhalten. Die Studierenden sollen in der Masterarbeit zeigen, dass sie in der Lage sind, innerhalb einer vorgegebenen Frist von sechs Monaten ein anspruchsvolles Problem der Angewandten Informatik selbstständig nach wissenschaftlichen Methoden zu bearbeiten. Neben den fachlichen Ergebnissen, werden im Rahmen der Masterarbeit auch fachübergreifende Qualifikationen erwartet. Studierende sollen die Projektorganisation selbstständig durchführen. Die Ergebnisse sind in schriftlicher Form zu dokumentieren. Dazu gehört auch der Vergleich ihrer Ergebnisse mit bereits vorhandenen Lösungen und Lösungsansätzen. Darüber hinaus sollen die Studierenden ihre Präsentations- und Argumentationsfähigkeit trainieren und verbessern.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Kritikfähigkeit • Projekt- und Zeitmanagement • Präsentation wissenschaftlicher Ergebnisse • Rhetorik und sprachliche Kompetenz 					

KAPITEL 2. MASTERARBEIT UND KOLLOQUIUM

- Fächerübergreifendes Denken und Arbeiten
- Wissenschaftliches Arbeiten und Schreiben

Inhalt:

Es soll eine anspruchsvolle Fragestellung der Angewandten Informatik bearbeitet und dokumentiert werden. Im Anschluss an die Bearbeitung der Masterarbeit werden die Ergebnisse in Form eines Kolloquiumsvortrags mit anschließender Diskussion präsentiert.

Lehrformen:

Praktikum/Projektarbeit

Prüfungsformen:

Abschlussarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 30 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Thomas Herrmann
Dozenten der RUB

Sonstige Informationen:

Im Normalfall sucht sich jede bzw. jeder Studierende nach eigenem Interesse und Neigung einen Lehrstuhl aus, an dem sie bzw. er die Masterarbeit schreiben möchte. Die meisten Lehrstühle veröffentlichen ihre angebotenen Themen. Oft werden Themen aber auch erst nach Absprache mit dem Studierenden gestellt, wobei Letzterer ein Vorschlagsrecht hat.

Die Anmeldung für die Masterarbeit erfolgt beim Prüfungsamt Angewandte Informatik. Für die Anmeldung ist das persönliche Erscheinen sowie die Vorlage des aktuellen Studentenausweises bzw. der Immatrikulationsbescheinigung notwendig.

Ingenieurinformatik

Design Optimization					
Modul-Nr: 129007	Credits: 6 CP	Workload: 180 h	Semester: 3 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Design Optimization			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse: JAVA-Grundkurs					
<p>Lernziele (learning outcomes): An important goal of this course is to present the theoretical foundations of mathematical optimization to students in a manner which allows them to use and employ design optimization for engineering applications in a sensible manner. This is achieved with a combination of theoretical lectures and practical exercises carried out using various computers software systems. In the second part of the course, students carry out team projects to solve engineering design tasks using their fundamental knowledge acquired in the first half of the course. Further, students must clearly present their projects results in a classroom setting to an audience with various technical background (the course is attended by students from applied computer science as well as by students from computational engineering). When the students have successfully complete this course, students</p> <ul style="list-style-type: none"> • will be familiar with the types of numerical algorithms available today to solve, in particular, advanced engineering tasks; • will be able to program software components to carry out design optimization tasks or employ engineering software systems to include design optimization aspects; • will have a good understanding of the basics of design optimization to be able to select proper optimization techniques in a given engineering situation and be able to implement efficient numerical solutions. 					
Inhalt:					
<ul style="list-style-type: none"> • Structural optimization as a tool for the optimal design of engineering tasks with respect 					

KAPITEL 3. INGENIEURINFORMATIK

to given quality objective functions, side constraints as well as inequality constraints.

- Development of optimization models for use in engineering applications
- Types of optimization categories (continuous, linear/non-linear, deterministic, simulation-based, multi-level, etc.)
- Strategies of optimization methods (classic indirect methods, direct numerical methods, global evolution strategies, partial swarm methods, distributed parallel methods, etc.)
- Software systems to implement design optimization tasks
- Application of design optimization paradigms to solve engineering tasks as a team project

Lehrformen:

lectures, exercises using computers, team projects

Prüfungsformen:

Completion of a team project (2-4 students); presentation of project results; oral examination

Voraussetzungen für die Vergabe von Kreditpunkten:

successful completion of a team project; presentation of project results in the classroom; oral examination

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Markus König
Prof. Dr. Markus König, Dr. Karlheinz Lehner

Design sozio-technischer Informationssysteme					
Modul-Nr: 260084	Credits: 5 CP	Workload: 150 h	Semester: 1-3 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung und Übung integriert			Kontaktzeit: 4 SWS	Selbststudium: 105 h	Gruppen- größe: 4-8
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse: Mensch-Maschine-Interaktion</p>					
<p>Lernziele (learning outcomes): Die Teilnehmer/innen sind in der Lage, bei der Entwicklung von informationstechnischen Lösungen frühzeitig den späteren Anwendungskontext und die Nutzungsbedingungen zu berücksichtigen. Sie könne helfen, die für die Systemeinführung und -Einsatz notwendigen organisatorischen Maßnahmen zu konzipieren.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Entwicklungsprozesse • Rechtliche Kompetenzen 					
<p>Inhalt: 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:</p> <ul style="list-style-type: none"> • Arbeitswissenschaft und Ergonomie • Psychologie • Organisation und Management • Rechtliche Aspekte • Betriebswirtschaftliche Aspekte 					

KAPITEL 3. INGENIEURINFORMATIK

- Kommunikationstheorie
- Datenschutz

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

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

Lehrformen:

Abwechslung von Vorlesungsinhalten, Seminarbeiträgen der Studierenden, Präsenzübungen und Erörterung der Projektbearbeitung

Prüfungsformen:

Seminarbeitrag, Präsentation der Bearbeitung einer Projektaufgabe in Kleingruppen, Prüfungsgespräch zu Überprüfung der Eigenleistung bei der Projektarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Ein Seminarbeitrag (25% der Endnote) und Präsentation der Projektarbeit (75%)

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Thomas Herrmann
Markus Jelonek Msc.

Sonstige Informationen:

Literatur:

Th. Herrmann (2012): Kreatives Prozessdesign. Springer-Gabler.

IT im Engineering					
Modul-Nr: 133640	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse: Keine</p>					
<p>Lernziele (learning outcomes):</p> <ul style="list-style-type: none"> • Allgemeine Übersicht der im gesamten Lifecycle eingesetzten IT-Lösungen • Systematische Erfassung von Engineeringprozessen • Darstellung von Prozessen in ereignisgesteuerten Prozessketten • Prozessanalyse • Erstellung von Konzepten für die Prozessoptimierung • Einführung von prozessunterstützenden IT-Lösungen <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Entwicklungsprozesse • Optimierungsmethoden 					
<p>Inhalt:</p> <ul style="list-style-type: none"> • Einführung in Anwendungssoftware-Systeme (z. B. CAx-Systeme, Engineering-Marktplätze, Portale, Kooperations-Anwendungen, etc.) in allen Phasen des Produktlebenszyklus. • Übersicht über Aufgaben und Funktionsweisen der Systeme in den Bereichen: <ul style="list-style-type: none"> – Produktentwicklung 					

KAPITEL 3. INGENIEURINFORMATIK

- Fertigung und Montage/Logistik
- Marketing und Vertrieb
- Service und Instandhaltung
- Einkauf und Beschaffung

- Vorstellung von integrierten Softwaresystemen (z. B. ERP-, CRM-, SCM-Systeme)
- Grundlagen der IT-Integration sowie der IT-Organisation und des IT-Managements
- Betreute Übungen in kleinen Gruppen am Rechner

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr.-Ing. Detlef Gerhard

Künstliche Intelligenz für Ingenieure					
Modul-Nr: 141005	Credits: 5 CP	Workload: 150 h	Semester: 1-3 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Künstliche Intelligenz für Ingenieure (141005)			Kontaktzeit: 4 SWS	Selbststudium: 94 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse: Diskrete Mathematik					
Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • haben Studierende fachspezifisches Grundlagenwissen der symbolischen Informationsverarbeitung und deren Umsetzung in Algorithmen, • sind die Studierenden in der Lage, typische Problemstellungen der Wissensverarbeitung im ingenieurstechnischen Bereich zu erkennen und zu strukturieren, • haben Studierende erste Erfahrungen im Umgang mit Sprachen der künstlichen Intelligenz durch Übungen im CIP-Pool gesammelt, • können die Studierenden die Möglichkeiten und Grenzen der Künstlichen Intelligenz diskutieren. 					
Inhalt: Grundprinzipien der Wissensrepräsentation und der symbolischen Informationsverarbeitung mit Anwendungsbeispielen aus der Automatisierungstechnik, insbesondere Suchverfahren in gerichteten Graphen, regelbasierte Systeme, Aufbau und Funktionsweise logikbasierter Systeme, Anwendungen für die Fehlerdiagnose in technischen Systemen.					
Lehrformen: Vorlesung und Praxisübungen					
Prüfungsformen: Klausurarbeit – 120 Minuten					
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur					
Verwendung des Moduls (in anderen Studiengängen): Masterstudiengang Elektrotechnik und Informationstechnik					
Stellenwert der Note für die Endnote: 5 / 112					
Modulbeauftragte/r und hauptamtlich Lehrende:					

Prof. Dr. Jan Lunze

Sonstige Informationen:

Literatur:

Lunze, Jan "Künstliche Intelligenz für Ingenieure - Methoden zur Lösung ingenieurtechnischer Probleme mit Hilfe von Regeln, logischen Formeln und Bayesnetzen", Oldenbourg Wissenschaftsverlag, 2010

Product Lifecycle Management					
Modul-Nr: 138577	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Product Lifecycle Management (PLM)			Kontaktzeit: V: 2 SWS; Ü: 2 SWS	Selbststudium: 120 h	Gruppengröße: beliebig
Teilnahmevoraussetzungen: Keine					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • verfügen Studierende über ein breites, integriertes Wissen über die Herausforderungen des ganzheitlichen Informationsmanagements im Produktlebenszyklus und die resultierenden Anforderungen an Softwaresysteme zur Unterstützung von PLM. • kennen und verstehen Studierende die Teilprozesse des Produktlebenszyklus, die Methoden des Product Lifecycle Management (PLM) sowie die wissenschaftlichen Grundlagen der zugrundeliegenden IT-Systeme. Indem sie praktische Beispiele und Aufgaben mit entsprechender Anwendungssoftware bearbeiten, können sie die erlernten Fertigkeiten im Umgang mit PLM Software auf konkrete und praxisorientierte PLM Problemstellungen übertragen. • haben Studierende ein umfassendes Verständnis vom Zusammenwirken der PLM Softwaresysteme und können kritisch die Eignung von Methoden zum Objektmanagement, zum Produktstruktur- und Konfigurationsmanagement sowie zum Projekt- und Prozessmanagement differenzieren und beurteilen. • können Studierende prozessorientiert an PLM Aufgabenstellungen herangehen, diese reflektieren und bewerten sowie selbstgesteuert verfolgen. • können Studierende kooperativ PLM-Aufgabenstellungen in heterogenen Gruppen bearbeiten, Abläufe und Ergebnisse begründen sowie über Sachverhalte umfassend kommunizieren. 					
<p>Inhalt: Die Veranstaltung vermittelt Methoden und Werkzeuge zum Product Lifecycle Management (PLM), insbesondere das dazu erforderliche Grundlagenwissen und die relevanten methodischen Aspekte von Produktinnovationsprozessen. Schwerpunkte bilden dabei die verschiedenen PLM-Funktionen entsprechender Softwaresysteme (z.B. Teile-, Dokumenten- und Produktstrukturmanagement, Klassifizierung, Konfigurationsmanagement, Projekt- und Prozessmanagement) Weiterhin werden allgemeine Methoden zur Organisation und Handhabung von Produktdaten und Benutzerinformationen sowie Methoden des Collaborative Engineering und die Vorgehensweise bei der PLM-Einführung vermittelt.</p>					

KAPITEL 3. INGENIEURINFORMATIK

Lehrformen: Vorlesungen, Übungen, Gruppenarbeiten, Moodle Unterstützung
Prüfungsformen: <ul style="list-style-type: none">• Schriftliche Klausur mit Fragen und Aufgaben zum Stoff der Vorlesung und der Übung - Dauer: 90 Minuten, Anteil der Modulnote: 100%
Voraussetzungen für die Vergabe von Kreditpunkten: 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
Verwendung des Moduls (in anderen Studiengängen): Studiengang Maschinenbau (Bachelor), Studiengang SEPM (Bachelor)
Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr.-Ing. Detlef Gerhard

Grundlagen der Verkehrsplanung und Verkehrstechnik					
Modul-Nr: IV-8b	Credits: 5 CP	Workload: 150 h	Semester: 1-3 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Grundlagen der Verkehrsplanung und Verkehrstechnik			Kontaktzeit: 4 SWS	Selbststudium: 90 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>Höhere Mathematik I und II</p>					
<p>Lernziele (learning outcomes): Das Modul soll die Studierenden mit den Grundlagen der Verkehrsplanung und der Straßenverkehrstechnik vertraut machen. Die Studierenden sollen lernen, Standardaufgaben aus diesem Bereich selbständig zu bearbeiten. Darüber hinaus sollen sie ein Grundverständnis für die verwendeten Methoden erlangen und Zusammenhänge innerhalb des Faches erkennen können. Sie sollen in die Lage versetzt werden, Vorgänge und Lösungen aus der Verkehrsplanung und Verkehrstechnik kritisch zu beurteilen.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • kennen Studierende einschlägige Methoden zur Erhebung und Analyse von Verkehrsdaten, • verstehen Studierende, wie das künftige Verkehrsgeschehen in einem Straßennetz anhand von Verkehrsdaten prognostiziert werden kann, • kennen Studierende elementare theoretische Grundlagen von Bewertungsmodellen und Simulationsverfahren in der Verkehrsplanung und Verkehrstechnik, • können die Studierenden Strecken von Autobahnen und Landstraßen sowie vorfahrtge-regelte und signalgesteuerte Knotenpunkte verkehrstechnisch bemessen und bewerten, • können die Studierenden die gängigen Richtlinien zur verkehrstechnischen Bemessung und zur Wirtschaftlichkeitsanalyse von Straßenverkehrsanlagen anwenden. 					
<p>Inhalt: Die Lehrveranstaltung behandelt das Basiswissen der Verkehrsplanung und der Straßenverkehrstechnik. Hierzu gehören:</p> <ul style="list-style-type: none"> • Verkehrsanalyse (Erhebungs- und Zählmethoden) • 4-Stufen-Algorithmus der klassischen Verkehrsplanung: <ul style="list-style-type: none"> – Verkehrserzeugungsmodelle und Prognoseverfahren – Verkehrsverteilung – Verkehrsaufteilung auf verschiedene Verkehrssysteme 					

KAPITEL 3. INGENIEURINFORMATIK

– Verkehrsumlegung auf die Strecken eines Netzes

- Kinematische Grundlagen der Verkehrstechnik
- Statistische Grundbegriffe, Warteschlangentheorie
- Verkehrsfluss auf Straßen, Fundamentaldiagramm
- Strecken von Autobahnen und Landstraßen
- Vorfahrtgeregelte Knotenpunkte
- Knotenpunkte mit Lichtsignalanlage
- Verkehrssicherheit
- Verkehrslärm
- Verfahren der Wirtschaftlichkeitsrechnung für die Infrastrukturplanung

Lehrformen:

Vorlesungen (2 SWS) und Übungen (2 SWS)

Prüfungsformen:

Modulklausur (90 min), optionale Hausarbeit zur Erlangung von Bonuspunkten für die Modulklausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Verwendung des Moduls (in anderen Studiengängen):

Pflichtmodul im Bachelor-Studiengang Umwelttechnik und Ressourcenmanagement sowie Teil eines Pflichtmoduls im Bachelor-Studiengang Bauingenieurwesen

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr.-Ing. Justin Geistefeldt

Sonstige Informationen:**Literatur:**

1. Schnabel, Lohse: Grundlagen der Straßenverkehrstechnik und der Verkehrsplanung, 3. Auflage, Beuth-Verlag
2. Steierwald, Künne, Vogt (Hrsg.): Stadtverkehrsplanung, 2. Auflage, Springer-Verlag
3. Köhler (Hrsg.): Verkehr – Straße, Schiene, Luft. Verlag Ernst und Sohn
4. Einschlägige Richtlinien und Merkblätter

Wissensbasierte Methoden					
Modul-Nr: 128701	Credits: 6 CP	Workload: 180 h	Semester: 2/4 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Wissensbasierte Methoden			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Objektorientierte Modellierung					
<p>Lernziele (learning outcomes): Neben den rein numerischen Berechnungsverfahren im Ingenieurwesen gibt es auch vielfältige Algorithmen, die nicht-numerischer Art sind und stattdessen mit symbolischen Größen Beziehungen zwischen den Komponenten eines Systems festhalten. Ein Verständnis solcher Verfahren soll geschult werden und den Studierenden befähigen solche Verfahren auch anzuwenden. Zu den behandelten Themen gehören neben KI-spezifisches Wissen wie Grundlagen zu Neuronalen Netzen oder zur Fuzzy Logik auch Kompetenzen wie analytisches und logisches Denken, fächerübergreifendes Denken und Arbeiten sowie die Fähigkeit zur Teamarbeit und Teamfähigkeit. Der Einsatz aktueller Hard- und Software zur Lösung von KI Problemen aus dem Bereich des Ingenieurwesens gibt einen Einblick in das breite Spektrum an Lösungsansätzen der KI. Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • kennen Studierende die Grundlagen verschiedener KI-Techniken wie Neuronale Netze oder der Fuzzy Logik bzw. die theoretischen Grundlagen der Aussage- und Prädikatenlogik; • verstehen Studierende durch Bearbeitung eines Team-Projekts aus dem Bereich der KI, wie sowohl die KI-Theorie als auch der Einsatz von KI-Software zur Lösung von Ingenieur Anwendungen erfolgreich eingesetzt werden kann. 					
<p>Inhalt:</p> <ul style="list-style-type: none"> • Historische Entwicklung und Bedeutung der wissensbasierten Methodik für moderne Entwurfs-, Planungs-, Konstruktions- und Fertigungsprobleme im Ingenieurwesen • Grundlegende Techniken der Repräsentation, Formalisierung und Verarbeitung von Ingenieurwissen (Aussagenlogik, Prädikatenlogik, Regelbasierte Systeme) • Methoden und Anwendungen der Fuzzy-Logik zur Erfassung unscharfer Sachverhalte im Ingenieurwesen und zur Modellierung komplexer technischer Systeme • AI-orientierte Programmiersprachen (Prolog, Jess) • Neuronale Netze • Integriertes Projekt aus dem Bereich der Fuzzy-Logik 					

KAPITEL 3. INGENIEURINFORMATIK

Lehrformen:

- Vorlesungen
- Übungen am Rechner und im Seminarraum
- Gruppenarbeiten

Prüfungsformen:

- Bearbeitung eines Projekts im Team (2-4 Studierende)
- Präsentation der Projekt-Ergebnisse
- Abgabegespräch

Voraussetzungen für die Vergabe von Kreditpunkten:

- Erfolgreiche Bearbeitung eines Projekts im Team
- Präsentation der Ergebnisse im Kurzvortrag
- Abgabegespräch

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Markus König
Prof. Dr. Markus König, Dr. Karlheinz Lehner

Grundlagen der automatischen Spracherkennung					
Modul-Nr: 141044	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung und Praxisübungen			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <ul style="list-style-type: none"> • Grundkenntnisse der digitalen Signalverarbeitung und der Wahrscheinlichkeitsrechnung • Grundlegende Programmierkenntnisse 					
<p>Lernziele (learning outcomes): Die Teilnehmer verstehen die theoretischen und praktischen Grundlagen automatischer Spracherkennungssysteme. Sie sind in der Lage, die Kernalgorithmen eines einfachen Spracherkenners selbstständig zu implementieren und verstehen die Prinzipien von aktuellen Erkennungssystemen für kleines und großes Vokabular. Dabei wird auch ein Verständnis für die Entwicklung von automatischen Mustererkennungsverfahren für ein breites Anwendungsfeld entwickelt.</p> <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten • Kritikfähigkeit • Selbständiges Lernen und Arbeiten • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Algorithmik • Künstliche Intelligenz und Maschinelles Lernen • Mathematik • Bild- und Signalverarbeitung • Programmieren 					

KAPITEL 3. INGENIEURINFORMATIK

Technologien: <ul style="list-style-type: none">• Python
Inhalt: <p>Die Vorlesung vermittelt Grundlagen und Algorithmen der maschinellen Spracherkennung in der Form, in der sie in aktuellen Systemen zur Erkennung fließender Sprache eingesetzt werden. Die folgenden Themen werden behandelt:</p> <ul style="list-style-type: none">• Grundlagen: Phonetik, Sprachwahrnehmung• Statistische Methoden: Klassifikation, Schätztheorie• Klassifikation mittels Deep Neural Networks• Merkmalsextraktion: Merkmale im Zeit- und Frequenzbereich, Cepstralanalyse• Spracherkennung mit Hidden Markov Modellen: Algorithmen, Modellinitialisierung, Baum-Welch-Reestimation, Numerische Aspekte, Systeme zur Einzel- und Verbundwörtererkennung, HMM/DNN-Systeme <p>Gleichzeitig werden in einem Programmierpraktikum die eingeführten Methoden angewandt.</p> <p>Die Übung ist projektorientiert; alle Übungsaufgaben zusammengenommen ergeben einen Verbundwörterkennner für fließend gesprochene Ziffernkettensätze. Dieser wird in Arbeitsgruppen von 2-3 Studenten erarbeitet.</p>
Lehrformen: <p>Vorlesung und Praxisübungen</p>
Prüfungsformen: <p>Klausur schriftlich, 120 Minuten</p>
Voraussetzungen für die Vergabe von Kreditpunkten: <p>Bestandene Modulabschlussklausur</p>
Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: <p>Prof. Dr.-Ing. Dorothea Kolossa</p>

Programmier- und Simulationstechnik

Fundamentals of GPU Programming					
Modul-Nr: 141374	Credits: 5 CP	Workload: 120 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 75 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: None Empfohlene Vorkenntnisse: C (programming language)					
Lernziele (learning outcomes): To learn how to program on graphics processing units (GPUs). Vermittelte Kompetenzen: Kernkompetenzen: <ul style="list-style-type: none"> • Selbständiges Lernen und Arbeiten Fachspezifische Kompetenzen: <ul style="list-style-type: none"> • Software-Entwurfsmethoden Technologien: <ul style="list-style-type: none"> • CUDA • C • Matlab 					
Inhalt: 1. GPU as a modern means for general-purpose massively parallel computations					

KAPITEL 4. PROGRAMMIER- UND SIMULATIONSTECHNIK

2. General GPU architecture and CUDA operational model
3. Basic CUDA syntax
4. Optimization strategies in GPU programming
5. Case study of general-purpose GPU programming

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Mündliche Prüfung, Übungsaufgaben

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Dr. Denis Eremin

Simulationstechnik					
Modul-Nr: 127012	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 105 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Objektorientierte Modellierung</p> <p>Empfohlene Vorkenntnisse: Keine</p>					
<p>Lernziele (learning outcomes): Dieses Modul vermittelt Kompetenzen zum Einsatz von rechnergestützten Simulationskonzepten zur Lösung praxisbezogener Aufgabenstellungen im Bau- und Umweltingenieurwesen. Im Rahmen der Übungsveranstaltungen erfolgt eine Einführung in aktuelle Simulations- und Optimierungssoftware. Die Projektarbeit wird als Gruppenarbeit durchgeführt.</p> <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Projekt- und Zeitmanagement • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Simulationsmethoden • Optimierungsmethoden • Programmieren <p>Technologien:</p> <ul style="list-style-type: none"> • AnyLogic • Java 					
<p>Inhalt: Es werden Vorgehensweisen zur Simulation komplexer Systeme vermittelt. Folgende Themen werden behandelt:</p> <ul style="list-style-type: none"> • Systemanalyse und Modellbildung 					

KAPITEL 4. PROGRAMMIER- UND SIMULATIONSTECHNIK

- System Dynamics
- Ereignisdiskrete Simulation
- Multiagentensimulation
- Aufbereitung von Eingangsdaten
- Stochastische Simulation
- Simulationsgestützte Optimierung
- Einführung in die Software AnyLogic

Innerhalb der Projektarbeit werden aktuelle Fragestellungen aus den Bau- und Umweltingenieurwissenschaften (Montage- und Logistikprozesse, Fußgängersimulation, Schadstoffausbreitung, etc.) aufgearbeitet und mit Hilfe einer objekt-orientierten Simulationssoftware analysiert. Den Studierenden werden Softwarelizenzen durch den Lehrstuhl zur Verfügung gestellt. Die Programmierung erfolgt unter Verwendung der Programmiersprache Java.

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Hausarbeit/Projektarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Markus König

Sonstige Informationen:**Literatur:**

1. Banks, J.; Carson II, J. S.; Nelson, B. L.; Nicol, D. M. (2005): Discrete-Event System Simulation, Pearson Prentice Hall
2. Bossel, H. (1994): Modellbildung und Simulation : Konzepte, Verfahren und Modelle zum Verhalten dynamischer Systeme, ein Lehr- und Arbeitsbuch, Vieweg Verlag
3. Biethahn, J.; Lackner, A.; Range, M; Brodersen, O. (2004): Optimierung und Simulation, Oldenbourg Verlag, München
4. Simulationssoftware AnyLogic der Firma XJ Technologies, <http://www.anylogic.com>

3D-Simulation in der Automatisierungstechnik					
Modul-Nr: 139050	Credits: 6 CP	Workload: 180 h	Semester: 1-4 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) 3D-Simulation in der Automatisierungstechnik			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppengröße: 20 Studierende (Übungen)
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Besuch der Vorlesung "Grundlagen der Automatisierungstechnik"</p>					
<p>Lernziele (learning outcomes):</p> <p>Nach einer allgemeinen Einführung in die Anforderungen und Möglichkeiten der 3D-Simulation in der Automatisierungstechnik erlernen die Studierenden die verschiedenen Schritte, die sich von der Idee über die Modellierung der Arbeitszelle und Programmierung der Automatisierungskomponenten bis zur Virtuellen Inbetriebnahme erstrecken. Ein besonderes Augenmerk wird in dieser Vorlesung auf die Industrielle Robotik gelegt, die in zahlreichen Beispielen und Anwendungen thematisiert wird.</p> <p>Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • beherrschen die Studierenden die Grundlagen der 3D-Simulationstechnik. • haben die Studierenden Fähigkeit erworben, automatisierungstechnische Aufgabenstellungen vorab über eine 3D-Simulation abzubilden und abzusichern. • kennen die Studierenden die wichtigsten Methoden und Softwaresysteme zur Lösung simulationstechnischer Probleme. 					
<p>Inhalt:</p> <p>Die Vorlesung deckt die folgenden Themenbereiche ab</p> <ul style="list-style-type: none"> • Simulation in der Automatisierungstechnik: Anforderungen und Möglichkeiten • Grafische 3D-Simulation • CAD-basierte Arbeitszellenmodellierung und 3D-Datenaustausch • Roboterprogrammierung • Offline-Programmierung und Virtuelle Inbetriebnahme • Grundlagen und Leistungsmerkmale von grafischen 3D-Simulationssystemen im industriellen Einsatz <p>Im Bereich der Kompetenzen im Kontext Digitalisierung ist besonders die Übung zu nennen, diese besteht aus der praktischen Umsetzung der von Modellierungs-, Programmierungs-</p>					

KAPITEL 4. PROGRAMMIER- UND SIMULATIONSTECHNIK

und Simulationsaufgaben mit einem kommerziell verfügbaren und industriell eingesetzten 3D-Robotersimulations- und Offline-Programmiersystem.
Lehrformen: Gemeinsame Vorlesung und Übung in kleineren Gruppen
Prüfungsformen: Klausur (90 Minuten)
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Verwendung des Moduls (in anderen Studiengängen): Masterstudiengang Maschinenbau
Stellenwert der Note für die Endnote: 5,36%
Modulbeauftragte/r und hauptamtlich Lehrende: Dr.-Ing. Alfred Hypki
Sonstige Informationen: Bei geringer Teilnehmerzahl kann die Prüfung auch mündlich (30 min.) angeboten werden.

Complexity Economics and Agent-based Modeling					
Modul-Nr: N/A	Credits: 10 CP	Workload: 300 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: 4 SWS	Selbststudium: 260 h	Gruppengröße: 30 Studierende aus den Wirtschaftswissenschaften und der Angewandten Informatik
<p>Teilnahmevoraussetzungen: Gute Englischkenntnisse Grundlegende Programmierkenntnisse</p> <p>Empfohlene Vorkenntnisse: Successful completion of a “Studienleistung”. Macroeconomics I and Macroeconomics II strongly recommended. Willingness to learn computer programming.</p>					
<p>Lernziele (learning outcomes): Das Modul verfolgt das Ziel, Studierende der Angewandten Informatik sowie der Wirtschaftswissenschaft in die Methode der agentenbasierten Modellierung und den dazugehörigen Simulationstechniken einzuführen. Hierbei lernen Studierende wie volkswirtschaftliche und sozialwissenschaftliche Fragestellungen entworfen, implementiert und analysiert werden. Im Rahmen der Seminararbeit sollen die Studierenden ein eigenes Modell implementieren und analysieren.</p> <p>Im Rahmen des Moduls erwerben Studierende folgende Kenntnisse:</p> <ol style="list-style-type: none"> 1) Grundlegende Kenntnisse der Komplexitätsökonomik und agentenbasierter Modellierung 2) Überführung von wissenschaftlichen Fragestellungen aus dem Bereich Volkswirtschaftslehre in agentenbasierte Computersimulationen 3) Praktische Arbeit mit agentenbasierten Modellen, Interpretation von Simulationsergebnissen und mögliche Grenzen des Ansatzes 4) Grundlegende Kenntnisse, um eigene Modelle zu implementieren und Simulationen selbstständig durchzuführen 5) Wissenschaftliches Schreiben 6) Die Programmiersprache NetLogo 					
<p>Inhalt: Komplexitätsökonomik ist eine neue Denkschule im Bereich der Volkswirtschaftslehre, der</p>					

zunehmend an Relevanz gewinnt. Hierbei wird eine Volkswirtschaft als ein komplexes, adaptives System betrachtet, das sich aus verschiedenen Agenten, z.B. Firmen, Haushalten und Banken, zusammensetzt. Im Vergleich zu bisherigen volkswirtschaftlichen Ansätzen sind die Agenten des Systems nicht (vollständig) rational und das System entwickelt sich aus den Interaktionen zwischen den (heterogenen) Agenten. Komplexitätsökonomik eignet sich sehr zur Untersuchung von sozialen Transformations- und Innovationsprozessen, wie der Digitalisierung.

In der Vorlesung wird Komplexitätsökonomik und die Methode der agentenbasierten Computersimulation vorgestellt und gezeigt, wie sie zur Analyse komplexer ökonomischer Systeme angewendet werden können. Im Rahmen der Übung werden diese Kenntnisse vertieft. Zudem wird geübt, wie agentenbasierte Modelle mithilfe NetLogo implementiert und simuliert werden.

Lehrformen:

Vorlesung und Übung

Prüfungsformen:

Seminararbeit, Klausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestehen der folgenden Leistungen:

- 1) Studienleistung (unbenotet)
- 2) Seminararbeit und Klausur (benotet)

Verwendung des Moduls (in anderen Studiengängen):

MSc Management and Economics, MSc Economic Policy Consulting

Stellenwert der Note für die Endnote: 10 / 112

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:

Will be announced at the beginning of the module.

Neuroinformatik

Autonomous Robotics: Action, Perception and Cognition					
Modul-Nr: 310501 (Übung: 310511)	Credits: 6 CP	Workload: 210 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 5 SWS	Selbststudium: 135 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Keine					
Lernziele (learning outcomes): We review experimental results in movement science, discuss mathematical models of movement generation, and use robotic instantiations of such models to illustrate their function. The mathematical language that pervades the theoretical work reviewed in the course comes from the theory of dynamical systems. The course includes tutorials on basic concepts in dynamical systems theory. The exercises provide opportunities to use those concepts in a variety of contexts. Another goal of the course is to expose students to interdisciplinary science. The exercises include readings of review papers in different relevant fields. An essay exercise practices reading and writing at the level of academic research papers. The course consists of a weekly 2-hour lecture, followed by a 1-hour exercise session. Exercise sheets given out each week must be handed in and individually corrected. They are discussed in the week after they are due.					
Vermittelte Kompetenzen: Kernkompetenzen:					
<ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Literaturrecherche und Dokumentation • Selbständiges Lernen und Arbeiten 					

KAPITEL 5. NEUROINFORMATIK

- Wissenschaftliches Arbeiten und Schreiben

Fachspezifische Kompetenzen:

- Mathematik
- Künstliche Intelligenz und Maschinelles Lernen

Inhalt:

Humans are the dexterous species. We excel at movement generation, in particular, at handling objects and generating the complex sequences of actions that achieve goals. This course looks at the fundamental processes of movement generation in humans and other animals and characterizes the special properties of human movement that emerge from the neural foundation. Object-oriented movement generation entails not only the timing and control of movement, but also object perception, scene representation, and the organization and planning of sequences. Movement generation thus cuts across a wide range of neural processes.

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur, Übungsaufgaben, Hausarbeit/Projektarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Gregor Schöner

Sonstige Informationen:

Eine Prüfung in 'Movement generation by Humans and Robots: A dynamical systems perspective' ist nicht möglich, wenn das ausgelaufene Modul 'Autonomous Robotics: Action, Perception, and Cognition' bereits erfolgreich bestanden wurde.

Literatur:

1. Valentino Braitenberg: Vehicles. Experiments in Synthetic Psychology, MIT Press, Cambridge, Mass 1984
2. Gregor Schöner, Michael Dose, Christoph Engels: Dynamics of behavior: Theory and applications for autonomous robotic architectures. Robotics and Autonomous Systems, 16:213-245 (1995)
3. Stephan K. U. Zibner, Christian Faubel, Ioannis Iossifidis, and Gregor Schöner: Dynamic Neural Fields as Building Blocks of a Cortex-Inspired Architecture for Robotic Scene Representation. IEEE Transactions Autonomous Mental Development 3:74-91 (2011)

Computational Neuroscience: Neural Dynamics					
Modul-Nr: 310001 (Übung: 310011)	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 135 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Keine					
Lernziele (learning outcomes):					
<ul style="list-style-type: none"> • Gain experience in interdisciplinarity bridging computer science and cognitive science. • Learn the concepts and methods of nonlinear dynamical systems in a concrete applied context. • Improve familiarity with methods of quantitative natural science, including measurement, graphing observables as a function of experimental control parameters and using models to interpret data. • Read scientific literature. 					
Vermittelte Kompetenzen:					
Kernkompetenzen:					
<ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Literaturrecherche und Dokumentation • Selbständiges Lernen und Arbeiten • Wissenschaftliches Arbeiten und Schreiben 					
Fachspezifische Kompetenzen:					
<ul style="list-style-type: none"> • Mathematik • Künstliche Intelligenz und Maschinelles Lernen 					
Inhalt:					
<p>This course provides an introduction into the theoretical cognitive and functional neurosciences from a particular theoretical vantage point, the dynamical systems approach. This approach emphasizes the evolution in time of behavioral and neural patterns as the basis of</p>					

KAPITEL 5. NEUROINFORMATIK

their analysis and synthesis. Dynamic stability, a concept shared with the classical biological cybernetics framework, is one cornerstone of the approach. Instabilities (or bifurcations) extend this framework and provide a basis for understanding flexibility, task specific adjustment, adaptation, and learning. The course includes tutorial modules that provide mathematical foundations. Theoretical concepts are expounded in reference to a number of experimental model systems which include the coordination of movement, postural stability, the perception of motion, and elementary forms of embodied cognition. In the spirit of Braitenberg's "synthetic psychology", autonomous robots are used to illustrate some of the ideas.

Exercises are integrated into the lectures. They consist of elementary mathematical exercises, the design of (thought) experiments and their analysis, and the design of simple artificial systems, all on the basis of the theoretical framework exposed in the main lectures.

One exercise takes the form of an essay for which participants read a scientific paper and answer questions in a longer illustrated text.

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur, Übungsaufgaben, Hausarbeit/Projektarbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 6 / 112**Modulbeauftragte/r und hauptamtlich Lehrende:**

Prof. Dr. Gregor Schöner

Sonstige Informationen:**Literatur:**

1. Martin Braun: Differential equations and their applications, Springer Verlag, New York, 1993
2. Gregor Schöner and Scott Kelso: Dynamic Pattern Generation in Behavioral and Neural Systems. Science 239: 1513-1520 (1988)
3. Gregor Schöner: Dynamical Systems Approaches to Cognition. In: The Cambridge Handbook of Computational Psychology,
4. Ron Sun, (ed.), Cambridge University Press (2008), pages 101-126

Computational Neuroscience: Vision and Memory					
Modul-Nr: 310504 (Übung: 310514)	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Computational Neuroscience: Vision and Memory			Kontaktzeit: 6 SWS	Selbststudium: 120 h	Gruppen- größe: 20
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>The mathematical level of the course is mixed but generally high. The tutorial is almost entirely mathematical. Mathematics required include calculus (functions, derivatives, integrals, differential equations, ...), linear algebra (vectors, matrices, inner product, orthogonal vectors, basis systems, ...), and a bit of probability theory (probabilities, probability densities, Bayes' theorem, ...).</p>					
<p>Lernziele (learning outcomes): After the successful completion of this course the students:</p> <ul style="list-style-type: none"> • know basic neurobiological facts about the visual system and the hippocampus, • know a number of related models and methods in computational neuroscience, • understand the mathematics of these methods, • can communicate about all this in English. 					
<p>Inhalt: This lecture covers basic neurobiology and models of selforganization in neural systems, in particular addressing vision (receptive fields, neural maps), and hippocampus (navigation and associative memory).</p>					
<p>Lehrformen: Flipped/inverted classroom</p>					
<p>Prüfungsformen: Oral exam, ca 20 min</p>					
<p>Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur</p>					
<p>Stellenwert der Note für die Endnote: 6 / 112</p>					
<p>Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Laurenz Wiskott</p>					

Sonstige Informationen:

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.

Machine Learning: Evolutionary Algorithms					
Modul-Nr: 310008 (Übung: 310018)	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung mit Integrierter Übung (flipped classroom)			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>The course is designed for Master students of the Angewandte Informatik program. The lecture Mathematics for Modeling and Data Analysis is not mandatory but recommended as a background.</p> <p>Participants should be familiar with linear algebra and elementary probability theory. For example, the following terms should be well known:</p> <ul style="list-style-type: none"> • vector, basis, linear map, matrix • norm, inner product, orthogonal • probability, distribution, density, quantile • normal distribution, expectation, variance, covariance <p>Several concepts from higher mathematics will appear during the course. These will be introduced on the fly.</p>					
<p>Lernziele (learning outcomes): Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • kennen die Teilnehmer die wichtigsten Klassen direkter Optimierungsverfahren und ihre algorithmischen Komponenten, • haben die Teilnehmer ein tiefes Verständnis evolutionärer Algorithmen, insbesondere für kontinuierliche Probleme, • kennen die Teilnehmer eine Reihe spezifischer Problemschwierigkeiten und die zugehörigen algorithmischen Komponenten, welche diese Adressieren, • können die Teilnehmer elementare Laufzeitanalysen durchführen und verstehen die wichtigsten Konvergenzklassen • können die Teilnehmer Optimierungsverfahren selbst implementieren und zur Lösung neuer Probleme anwenden. 					

KAPITEL 5. NEUROINFORMATIK

Inhalt: Breiter Überblick über Optimierungsverfahren. Evolutionäre Optimierungsverfahren für black-box Optimierungsverfahren Algorithmische Komponenten für schlechte Konditionierung, Multimodalität, Rauschen, Nebenbedingungen und Mehrzieloptimierung. Konvergenz- und Laufzeitanalyse.
Lehrformen: flipped classroom
Prüfungsformen: Klausur 90 Minuten, Wiederholungsklausur nach 6 Monaten
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Verwendung des Moduls (in anderen Studiengängen): Master Medizinphysik Master Physik Sämtliche Ingenieurstudiengänge Die Veranstaltung wird auch regelmäßig von Studierenden sämtlicher Ingenieurstudiengänge, der Chemie, der Mathematik, aus cognitive science, von Doktoranden, sowie von Studierenden der TU Dortmund (insbesondere automation and robotics) nachgefragt.
Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Jun.-Prof. Dr. Tobias Glasmachers

Master-Praktikum: Autonomous Robotics					
Modul-Nr: 310036 (WS), 310536 (SS)	Credits: 3 CP	Workload: 90 h	Semester: beliebiges Sem.	Turnus: immer	Dauer: 1 Semester
Lehrveranstaltungen: Praktikum/Projektarbeit			Kontaktzeit: 2 SWS	Selbststudium: 60 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse: Vorlesung Autonomous Robotics: Action, Perception, and Cognition</p>					
<p>Lernziele (learning outcomes):</p> <ul style="list-style-type: none"> • Einüben des praktischen Arbeitens im Umfeld der autonomen Robotik, insbesondere der Entwurf von mathematischen Verfahren und ihre algorithmische Implementation zur Erzeugung von zeitlich strukturierten Abläufen • Einüben der Dokumentation von technischen Programmieraufgaben. • Verständnis der Grundlagen der autonomen Robotik anhand des einfachsten Robotersystems, eines autonomen Kleinstvehikels. • Einüben des Lesens und Schreibens von wissenschaftlich-technischen Artikeln <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Literaturrecherche und Dokumentation • Selbständiges Lernen und Arbeiten • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Programmieren <p>Technologien:</p> <ul style="list-style-type: none"> • Matlab 					
<p>Inhalt:</p> <p>Ziel des Praktikums ist es, die Grundlagen der autonomen Robotik anhand von Experimenten zu vermitteln, die eine enge Kopplung an aktuelle wissenschaftliche Probleme dieses Feldes aufweisen. Am Beispiel der Kleinstroboter vom Typ E-Puck wird hierbei die Erzeugung</p>					

KAPITEL 5. NEUROINFORMATIK

und Organisation von künstlichem Verhalten behandelt. Unter Verwendung der robotereigenen Sensorik werden experimentelle Aufgaben durch Implementierungen in der Programmierumgebung MATLAB gelöst. Der theoretische Hintergrund der Lösungen ist der dynamische Systeme Ansatz der kognitiven Robotik.

Eine optionale Erweiterung des Praktikums durch das der dritte CP erzielt wird, ist das Lesen einer wissenschaftlichen Veröffentlichung im Umfeld der im Praktikum behandelten Methoden und das Schreiben eines Essays, in dem Fragen zu dem Artikel durch einen längeren, strukturierten und illustrierten Text beantwortet werden.

Lehrformen:

Praktikum/Projektarbeit

Prüfungsformen:

Übungsaufgaben

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 3 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Gregor Schöner

Mitarbeiter des Instituts

Sonstige Informationen:**Literatur:**

1. Valentino Braitenberg: Vehicles. Experiments in Synthetic Psychology, MIT Press, Cambridge, Mass 1984
2. Estela Bicho and Gregor Schöner: The dynamical approach to autonomous robotics demonstrated on a low-level vehicle platform. Robotics and Autonomous Systems 21:23-35 (1997)

Computer Vision: Deep Learning Lab Course					
Modul-Nr: 310531	Credits: 2 CP	Workload: 60 h	Semester: 2 / 4 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Computer Vision: Deep Learning Lab Course			Kontaktzeit: 1-wöchiger Blockkurs keine SWS	Selbststudium: (ganztägige Präsenzveranstaltung) h	Gruppengröße: 10 - 20 Studierende
Teilnahmevoraussetzungen: Familiarity with at least one imperative programming language, preferably Python					
Empfohlene Vorkenntnisse: Basic command of Python					
Lernziele (learning outcomes): The lab course focuses on the practical aspects of implementing deep learning models for large-scale image classification tasks. Students that have finished the course will be able to implement own machine learning models and training environments (including regularization and data augmentation techniques) in Tensorflow and will have the necessary background to tackle other computer vision or machine learning tasks in self-study using tensorflow or scikit-learn.					
Inhalt: <ul style="list-style-type: none"> • Basic operations of computer vision and machine learning • Definition of simple deep learning models in Tensorflow • Methods for improving training in Tensorflow 					
Lehrformen: Der Blockkurs beginnt täglich mit einer Vorlesung, die in die Thematik des Tages einführt, gefolgt von praktischen Programmierübungen für den Rest des Tages, in denen die Thematik vertieft wird. Diese Aufgaben können allein oder in kleinen Gruppen bearbeitet werden. Zur Kontrolle sollen die Teilnehmer das von ihnen entwickelte Programm vorstellen und Grenzfälle aufzeigen. Ein Tag des Kurses ist zum Aufholen nicht fertiggestellter Programmieraufgaben vorgesehen. An diesem Tag werden keine neuen Themen eingeführt.					
Prüfungsformen: Bewertung einer zusammenfassenden schriftliche Ausarbeitung im Umfang von 3 – 5 Seiten					
Voraussetzungen für die Vergabe von Kreditpunkten: Bewertung der schriftlichen Ausarbeitung mit mindestens 50%					
Stellenwert der Note für die Endnote: 2 / 112					

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Sebastian Houben
Sebastian Houben und Mitarbeiter des Instituts

Sonstige Informationen:

Limited number of participants. This is a one-week course with full-day attendance.

Computer Vision: Deep Learning					
Modul-Nr: 310005	Credits: 6 CP	Workload: 180 h	Semester: 1 / 3 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung und Übungen Computer Vision: Deep Learning			Kontaktzeit: 4 SWS	Selbststudium: 1 h	Gruppen- größe: 30 - 50 Studie- rende
Teilnahmevoraussetzungen: Keine					
<p>Lernziele (learning outcomes): The participants are able to understand, implement, and extend the main deep learning approaches with applications in computer vision. As many of the concepts are very recent research results, an emphasis is put on working with recent research papers and results that elucidate but not fully explain certain typical phenomena from the field. The practical exercises will enable students to implement and train simple deep learning models and develop methods for debugging, extended tooling and self-study.</p>					
<p>Inhalt:</p> <p>Inhalt:</p> <ul style="list-style-type: none"> • Notions, basic techniques, and key problems in machine learning • Defining and training deep neural network models • Backpropagation • Nuts and bolts in training deep neural networks • Hyperparameter optimization • Tensorflow • Convolutional Neural Networks • Object detection and image segmentation • Visualizing and understanding deep neural networks • Generative Models 					
<p>Lehrformen: 2 stündiges Vorlesungsformat mit zusätzlich wöchentlich zu bearbeitenden praktischen Übungen in Form von mathematischen Herleitungen, Verständnisfragen und Programmimplementierungen Vorlesungen und Übungen werden je nach Publikum auf Englisch abgehalten.</p>					
<p>Prüfungsformen: Klausur von 90 – 120 Minuten Länge mit mathematischen Herleitungen und Verständnisfragen</p>					
<p>Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur</p>					

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Jun. Prof. Dr. Sebastian Houben

Kryptologie und Theoretische Informatik

Kryptanalyse 1 (Einführung in die asymmetrische Kryptanalyse)					
Modul-Nr: 150314	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 4 SWS	Selbststudium: 90 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse: Inhalte der Vorlesungen Einführung in die Kryptographie 1 und 2.</p>					
<p>Lernziele (learning outcomes): Die Studierenden beherrschen die grundlegenden Algorithmen in der Kryptanalyse.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • IT-Sicherheit • Algorithmik • Mathematik 					
<p>Inhalt: Die Vorlesung gibt einen Einblick in grundlegende Methoden der Kryptanalyse. Der Stoff-</p>					

KAPITEL 6. KRYPTOLOGIE UND THEORETISCHE INFORMATIK

plan umfasst die folgenden Themen:

- Brute Force und Geburtstagsangriffe
- Time-Memory Tradeoffs
- Seitenkanalangriffe
- Gittertheorie und der LLL-Algorithmus
- Gitterbasierte Angriffe auf RSA
- Hidden Number Problem und Angriffe auf DSA
- Faktorisieren mit Faktorbasen
- Diskreter Logarithmus, Index-Calculus

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Alexander May

Sonstige Informationen:

Diese Veranstaltung wird mit 4+2 SWS in der ersten Hälfte des Semesters angeboten.

Literatur:

Skript zur Vorlesung.

Kryptanalyse 2 (Asymmetrische Kryptanalyse)					
Modul-Nr: 150334	Credits: 4 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Wintersemester (NICHT im WS 17/18!)	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 105 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Inhalte der Vorlesungen Einführung in die Kryptographie 1 und 2 sowie Einführung in die asymmetrische Kryptanalyse.</p>					
<p>Lernziele (learning outcomes): Die Studierenden beherrschen die wichtigsten Algorithmen in der Kryptanalyse.</p> <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • IT-Sicherheit • Algorithmik • Mathematik 					
<p>Inhalt:</p> <p>Die Vorlesung gibt einen Einblick in fortgeschrittene Methoden der Kryptanalyse. Der Stoffplan umfasst die folgenden Themen:</p> <ul style="list-style-type: none"> • Pollards p-1 Methode • Faktorisieren mit Elliptischen Kurven • Pohlig-Hellman Algorithmus • Cold-Boot Angriffe und Fehlerkorrektur von Schlüsseln • Generalisiertes Geburtstagsproblem • Lösen von polynomiellen Gleichungssystemen mit Gröbnerbasen 					

KAPITEL 6. KRYPTOLOGIE UND THEORETISCHE INFORMATIK

- Hilbert Basissatz und Buchberger Algorithmus
- Fourier und Hadamard Walsh Transformation

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 4 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Alexander May

Sonstige Informationen:

Diese Veranstaltung wird mit 4+2 SWS in der zweiten Hälfte des Semesters angeboten.

Literatur:

Skript zur Vorlesung.

Algorithmische Geometrie					
Modul-Nr: 150341 (Übung: 150342)	Credits: 6 CP	Workload: 180 h	Semester: 1-3 Sem.	Turnus: Jedes zweite Semester, idR im WiSe	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: 20
<p>Teilnahmevoraussetzungen: Grundlegende Kenntnisse ueber Algorithmen und Datenstrukturen.</p> <p>Empfohlene Vorkenntnisse: Grundlagen der Stochastik.</p>					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls</p> <ul style="list-style-type: none"> • kennen Studierende grundlegende geometrische Algorithmen und Datenstrukturen • können Studierende Algorithmen nach dem Sweep-Paradigma analysieren und entwerfen • können Studierende inkrementelle Algorithmen entwerfen und analysieren, insbesondere randomisiert inkrementelle Algorithmen • können Studierende geometrische Algorithmen nach dem Teile-und-Herrsche Prinzip analysieren und entwerfen • können Studierende für Bereichsanfragen geeignete Datenstrukturen aussuchen 					
<p>Inhalt: Die Algorithmische Geometrie beschäftigt sich mit dem Entwurf und der Analyse von Algorithmen und Datenstrukturen für geometrische Probleme. Dazu werden zunächst einige grundlegende Probleme betrachtet, wie das Berechnen der konvexen Hülle einer Punktmenge, der Schnittpunkte einer Menge von Strecken oder einer Triangulierung eines einfachen Polygons. Anschließend sehen wir Algorithmen zum Berechnen bekannter geometrische Strukturen, wie das Voronoi-Diagramm, die Delaunay-Triangulierung und Arrangements. Ebenfalls betrachten wir Datenstrukturen für effiziente Anfragen auf geometrischen Daten, wie Rangetrees, kd-Bäume und Quadtrees. Dabei kommen vor allem drei Arten von Algorithmen zum Einsatz: inkrementell, teile-und-herrsche, und sweep. Manche von diesen treten als randomisierte Algorithmen auf.</p>					
<p>Lehrformen: Vorlesung als kombinierter Folien- und Tafelvortrag, Übungen mit begleitendem Implementierungsprojekt</p>					

Prüfungsformen: Mündliche Prüfung, Abgabe von schriftlichen Übungen, Abgabe eines Projektberichtes
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene mündliche Prüfung, sowie erfolgreiches Bearbeiten der Übungen und des Projektes
Verwendung des Moduls (in anderen Studiengängen): Angewandte Informatik, Informatik, Mathematik
Stellenwert der Note für die Endnote: 6 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Maike Buchin
Sonstige Informationen: Literatur: Die Vorlesung orientiert sich groesstenteils an dem Buch "Computational Geometry: Algorithms and Applications", von Mark de Berg, Otfried Cheong, Marc van Kreveld, und Mark Overmars (3te Auflage, 2008, Springer).

Kryptographie					
Modul-Nr: 150312 (Übung: 150313)	Credits: 9 CP	Workload: 270 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 6 SWS	Selbststudium: 180 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Keine					
Lernziele (learning outcomes): Die Vorlesung bietet eine Einführung in moderne Methoden der symmetrischen und asymmetrischen Kryptographie. Dazu wird ein Angreifermodell definiert und die Sicherheit der vorgestellten Verschlüsselungs-, Hash- und Signaturverfahren unter wohldefinierten Komplexitätsannahmen in diesem Angreifermodell nachgewiesen.					
Vermittelte Kompetenzen: Kernkompetenzen:					
<ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten 					
Fachspezifische Kompetenzen:					
<ul style="list-style-type: none"> • IT-Sicherheit • Algorithmik • Mathematik 					
Inhalt:					
<ul style="list-style-type: none"> • Sichere Verschlüsselung gegenüber KPA-, CPA- und CCA-Angreifern • Pseudozufallsfunktionen und -permutationen • Message Authentication Codes • Kollisionsresistente Hashfunktionen • Blockchiffren • Konstruktion von Zufallszahlengeneratoren • Diffie-Hellman Schlüsselaustausch • Trapdoor Einwegpermutationen 					

KAPITEL 6. KRYPTOLOGIE UND THEORETISCHE INFORMATIK

- Public Key Verschlüsselung: RSA, ElGamal, Goldwasser-Micali, Rabin, Paillier
- Einwegsignaturen
- Signaturen aus kollisionsresistenten Hashfunktionen
- Random-Oracle Modell

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

Klausur

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 9 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Alexander May

Prof. Dr. Alexander May, Prof. Dr. Eike Kiltz

Sonstige Informationen:

Diese Veranstaltung ist im Vorlesungsverzeichnis der Mathematik als "Kryptographie I + II" aufgeführt.

Literatur:

Katz, Lindell, "Introduction to Modern Cryptography", Chapman und Hall/CRC, 2008

Kryptographische Protokolle					
Modul-Nr: 150343 (Übung: 150344)	Credits: 5 CP	Workload: 270 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 225 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Kryptographie</p> <p>Empfohlene Vorkenntnisse: Keine</p>					
<p>Lernziele (learning outcomes): Entwicklung eines tieferen Verständnis für die unter "Inhalt" genannten Themen.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten • Selbständiges Lernen und Arbeiten • Wissenschaftliches Arbeiten und Schreiben <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Algorithmik • IT-Sicherheit • Mathematik 					
<p>Inhalt: Die Vorlesung beschäftigt sich mit erweiterten kryptographischen Protokollen und deren Anwendungen. Themenübersicht:</p> <ul style="list-style-type: none"> • Identity-based Encryption • Digital Signatures • Secret sharing • Threshold Cryptography • Secure Multiparty Computation 					

KAPITEL 6. KRYPTOLOGIE UND THEORETISCHE INFORMATIK

Lehrformen: Vorlesung (+Übung)
Prüfungsformen: Klausur, Mündliche Prüfung, Übungsaufgaben
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Stellenwert der Note für die Endnote: 5 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Eike Kiltz

Theorie des maschinellen Lernens					
Modul-Nr: 150338 (Übung: 150339)	Credits: 9 CP	Workload: 270 h	Semester: 2 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung Theorie des maschinellen Lernens b) Übungen zur Vorlesung			Kontaktzeit: 6 SWS	Selbststudium: 180 h	Gruppengröße: in der Regel maximal 20 Studierende
Teilnahmevoraussetzungen: Grundkenntnisse in Mathematik und theoretischer Informatik					
Empfohlene Vorkenntnisse: Vorlesung über Theoretische Informatik					
Lernziele (learning outcomes): Die Studierenden werden mit mathematischen Modellen für das maschinelle Lernen vertraut gemacht. Sie erwerben die Fähigkeit, Lernalgorithmen zu beurteilen und zu vergleichen anhand des Grades, in welchem diese (exakt beschriebene) Erfolgskriterien erreichen. Sie erwerben Techniken sowohl zum Design effizienter Lernalgorithmen als zum Nachweis der inhärenten Härte eines Problems. Nach dem erfolgreichen Abschluss des Moduls					
<ul style="list-style-type: none"> • kennen Studierende die wichtigsten Lernmaschinen (wie zum Beispiel Support Vector Machines und verwandte Modelle), • verstehen Studierende den Unterschied zwischen empirischer und realer Fehlerrate und kennen Techniken zum Umgang mit dem Problem des overfitting der Daten (mit einem zu komplexen Modell), • können Studierende zwischen uniformer und nicht uniformer Lernbarkeit einer Hypothesenklasse unterscheiden und kennen die dazu passenden Theorien und Lernregeln. 					
Inhalt: Gegenstand der Vorlesung ist die statistik-basierte Theorie des maschinellen Lernens. Insbesondere wird die Methode der strukturierten Risikominimierung vermittelt sowie die ihr zugrunde liegenden statistischen Lehrsätze. Es werden sowohl Techniken zum Entwurf effizienter Lernalgorithmen besprochen als auch informations- oder berechnungstheoretische Barrieren, die bestimmte Lernprobleme als nicht effizient lösbar erscheinen lassen.					
Lehrformen: Vorlesung mit begleitenden Übungen					
Prüfungsformen: mündliche Prüfung mit einer Dauer von 25-30 Minuten					

Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Verwendung des Moduls (in anderen Studiengängen): Studiengang Mathematik
Stellenwert der Note für die Endnote: 9 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Hans Ulrich Simon
Sonstige Informationen: Literatur: Shai Shalev-Shwartz und Shai Ben-David, "Understanding Machine Learning - From Theory to Algorithms", Cambridge University Press

Betriebssystemsicherheit					
Modul-Nr: 141342	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 4 SWS	Selbststudium: 90 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Erfahrung in systemnaher Programmierung sowie der Programmiersprache C sind hilfreich für das Verständnis der vermittelten Themen.</p>					
<p>Lernziele (learning outcomes): Die Studierenden beherrschen theoretische und praktische Aspekte der Sicherheit von Betriebssystemen und sind zu einer kritischen Betrachtung der Systemsicherheit in der Lage.</p>					
<p>Inhalt:</p> <p>Im ersten Teil der Veranstaltung werden verschiedene Sicherheitsaspekte von Betriebssystemen vorgestellt und erläutert. Dazu werden sowohl wichtige Angriffsmethoden (z.B. Buffer Overflows oder Race Conditions) als auch Abwehrstrategien (z.B. nicht-ausführbarer Speicher oder Address Space Layout Randomization) diskutiert. Andere Themen, die im Mittelpunkt dieses Teils der Vorlesung stehen, sind Virtualisierung/Hypervisor sowie das sogenannte Einsperrungs-Problem (Confinement Problem) und die damit verbundene Analyse der verdeckten Kanäle in einem Computer-System.</p> <p>Im zweiten Teil der Veranstaltung liegt der Schwerpunkt auf Schadsoftware. Dazu werden zunächst die Grundbegriffe in diesem Bereich erläutert und danach verschiedene Methoden zur Erkennung von Schadsoftware diskutiert. Wichtige Algorithmen in diesem Bereich werden vorgestellt und verschiedene Ansätze für Intrusion Detection Systeme werden behandelt.</p> <p>Im praktischen Teil der Veranstaltung wird die Sicherheit von mehreren realen Systemen analysiert. Ein integraler Teil der Veranstaltung sind die Übungen, die den Stoff mit praktischen Beispielen veranschaulichen und vertiefen.</p>					
<p>Lehrformen: Vorlesung (+Übung)</p>					
<p>Prüfungsformen: Klausur</p>					
<p>Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur</p>					
<p>Stellenwert der Note für die Endnote: 5 / 112</p>					
<p>Modulbeauftragte/r und hauptamtlich Lehrende:</p>					

Operations Research

Industrielles Kundenmanagement					
Modul-Nr: 139010	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Industrielles Kunden- Management			Kontaktzeit: 4 SWS	Selbststudium: 105 h	Gruppen- größe: 20 Studierende
Teilnahmevoraussetzungen: Keine					
<p>Lernziele (learning outcomes): Zielsetzung: Die Studierenden erwerben fachspezifische Kompetenzen in Form von Kenntnissen zu Grundlagen von an den Maschinenbau angrenzenden, relevanten Ingenieurwissenschaften, wie dem Wirtschaftsingenieurwesen und darüber hinaus Ansätze der Wirtschaftspsychologie und wirtschaftswissenschaftliche Aspekte. Die Studierenden sollen zentrale Methoden des Industriellen Kunden-Managements in den relevanten Management-Komponenten aktiv anwenden können und theoriegeleitet wissenschaftliche (Fall-)Studien bearbeiten, kritisch darstellen und präsentieren. Als generische Kompetenzen werden die Studierenden zu vernetztem Denken angeregt und in die Lage versetzt etablierte Methoden und Verfahren auszuwählen und anzuwenden und daraus kritisches Managementhandeln abzuleiten. Dies beinhaltet spezifische Kenntnisse und instrumentale Kompetenzen des industriellen Kunden-Managements aus prozessualer, organisatorischer und strategischer Sicht. Die Studierenden erlangen vertiefte, interdisziplinäre instrumentale Kompetenzen und können diese situativ anwenden. Dabei lernen die Studierenden die Grundlagen, Methoden und Verfahren der wirtschaftsingenieur- und verhaltenswissenschaftlichen Ansätze in Praxis und Forschung im Bereich des IKM kennen. Dies beinhaltet die Analyse, Bewertung und Gestaltung der Aufbau- und Ablauforganisation und der Geschäftsmodelle unterschiedlicher Organisationen sowie wesentliche Elemente der kundenorientierten Strategieentwicklung im Kontext einer digitalen Transformation der beruflichen und privaten Lebenswelt.</p>					
<p>Inhalt: In den Veranstaltungen zum Industriellen Kunden-Managements werden Theorien, Modelle und Ansätze dargestellt und anhand von Fallstudien und Business-Episoden vertieft und reflektiert. Die Veranstaltung orientiert sich an der zyklischen Bearbeitung von 12 wesentlichen</p>					

KAPITEL 7. OPERATIONS RESEARCH

Management-Kompetenzfeldern zum ganzheitlichen Kunden-Management.

12 Episoden bzw. Management-Kompetenzfelder:

1. Innovationsmanagement
2. Technologiemanagement
3. FundE Management
4. Prozessmanagement
5. Qualitätsmanagement
6. Produktmanagement
7. Marketingmanagement
8. Vertriebsmanagement
9. Servicemanagement
10. Projektmanagement
11. Change Management
12. Kundenmanagement

Arbeitsaufwände: - Präsenzzeit: 45 h Präsenzstudium - Vor und Nachbereitung (einschl. Prüfung): 105 h Eigenstudium

Lehrformen:

Unterricht in Seminarform, Gruppenarbeiten, Bearbeitung von Fallbeispielen

Prüfungsformen:

Gruppenarbeit mit Präsentation zur Erlangung von Bonuspunkten (freiwillig); Mündliche Abschlussprüfung in Kleingruppen von 3-4 Personen und einer Dauer von 20 Minuten pro Prüfling

Voraussetzungen für die Vergabe von Kreditpunkten:

Teilnahme und bestehen der mündlichen Prüfung

Verwendung des Moduls (in anderen Studiengängen):

Studiengang Maschinenbau Master; Optionalbereich

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. phil. Joachim Zülch
Dr.-Ing. Matthias Bartels

Sonstige Informationen:

Das Modul eignet sich für interessierte Studierende der Master-Phase im Optionalbereich und

KAPITEL 7. OPERATIONS RESEARCH

erfordert keine speziellen Vorkenntnisse. Weitere Informationen sowie die Unterlagen zu Vorlesungen und Übungen werden über Moodle zur Verfügung gestellt.

Rationales Entscheiden					
Modul-Nr: 074210 (Übung: 074211)	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Letztmalig im WS 19/20!	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 3 SWS	Selbststudium: 105 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Die Kenntnisse linearer Optimierungsmodelle und statistischer Grundlagen wird vorausgesetzt. Eine Auffrischung der Kenntnisse erfolgt im Rahmen eines Kolloquiums (1 SWS).</p> <p>Empfohlene Vorkenntnisse: Betriebswirtschaftliche Grundkenntnisse sind hilfreich.</p>					
<p>Lernziele (learning outcomes): Absolventen dieser Veranstaltung sind in der Lage, komplexe Entscheidungssituationen zu strukturieren und selbständig entscheidungstheoretisch fundierte Vorschläge für gute Lösungen zu entwickeln. Dieses allgemeine methodisch orientierte betriebswirtschaftliche Modul eignet sich besonders als theoretische Vorbereitung für die analytische strukturierte Behandlung komplexer Probleme.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten • Selbständiges Lernen und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Optimierungsmethoden • Wirtschaftliches Handeln 					
<p>Inhalt: Es werden grundlegende entscheidungs- und informationstheoretische Konzepte vorgestellt und anhand wirtschafts-wissenschaftlicher Beispiele verdeutlicht. Besonders berücksichtigt werden Entscheidungen unter Unsicherheit, also Risiko und Ungewissheit, im Unterschied zu solchen unter Sicherheit. Weiter werden Entscheidungen mit mehreren Zielen oder mit mehreren Entscheidungsträgern ausführlich behandelt. Dynamische Entscheidungen und weiterführende Ansätze werden ebenfalls vorgestellt.</p>					
<p>Lehrformen: Vorlesung (+Übung)</p>					

KAPITEL 7. OPERATIONS RESEARCH

Prüfungsformen: Klausur
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Stellenwert der Note für die Endnote: 5 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Brigitte Werners
Sonstige Informationen: Letztmalig im WS 19/20 Literatur: Veranstaltungsunterlagen und Literaturhinweise werden über Blackboard bereitgestellt

Bioinformatik

Bioinformatik in der Proteomik I					
Modul-Nr: 201911	Credits: 5 CP	Workload: 150 h	Semester: 1-4 Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: V: 2 SWS; Ü: 1 SWS	Selbststudium: 105 h h	Gruppen- größe: 3 bis 20 Studie- rende
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Grundlegende Programmierkenntnisse, Statistik-Grundkenntnisse					
Lernziele (learning outcomes): Fachspezifische Kompetenzen: Nach dem erfolgreichen Abschluss dieses Moduls					
<ul style="list-style-type: none"> • haben die Studierenden grundlegende Sachverhalte der Proteinbiochemie kennengelernt, • sie können die Prinzipien der Massenspektrometrie als Schlüsseltechnologie der Proteomik erklären, • sie können aktuelle Methoden der Bioinformatik der Proteomik, die zur Analyse von Rohdaten (Massenspektren) mit dem Ziel der Peptid-/Protein-Identifizierung und -Quantifizierung verwendet werden, erläutern, • sie verstehen die diesen Methoden zugrundeliegenden algorithmischen und statistischen Ideen, • sie können Proteomik-spezifische Software und die Workflow-Engine KNIME nutzen, • sie können eigene Lösungsideen entwerfen und programmieren • und sie können die besprochenen Software-Tools und Methoden auf reale Daten und Fragestellungen anwenden. 					

KAPITEL 8. BIOINFORMATIK

Fachübergreifende/generische Kompetenzen:

- instrumentale Kompetenzen: - Intensive Nutzung der Lernplattform Moodle
- systemische Kompetenzen: - Selbständiges Lernen und Arbeiten - Teamarbeit und Teamfähigkeit
- kommunikative Kompetenzen: - Präsentation der eigenen Arbeit und Ergebnisse - Vermittlung der in der Bioinformatik angewandten englischen Fachbegriffe - Rhetorik und sprachliche Kompetenz (deutsch / englisch)

Vermittelte Kompetenzen:

Kernkompetenzen:

- Selbständiges Lernen und Arbeiten
- Teamarbeit und Teamfähigkeit

Inhalt:

- Grundlagen der Proteinbiochemie
- Eigenschaften von Aminosäuren
- Grundlagen der Massenspektrometrie
- Rohdaten-Verarbeitung
- Protein-Datenbanken
- Tryptische Peptide und In silico-Verdau
- Prinzipien von Spektren-Identifizierungs-Suchmaschinen und -scores
- Beschränkung der False Discovery Rate mit dem Decoy-Ansatz
- PSM-spezifische Score-Korrektur (Percolator)
- Protein-Inferenz
- Protein-Quantifizierung
- Vorverarbeitung quantitativer Daten
- Qualitätskontrolle
- Software-Tools im Bereich Bioinformatik der Proteomik (Übung)
- Praktische (Programmier-)Aufgaben (Übung)
- Workflow-Engine KNIME (Übung)

Lehrformen:

Vorlesung: Folien-basierte Vorlesung. Übung: Lösung von kleinen praktischen Problemen

KAPITEL 8. BIOINFORMATIK

als Hausaufgaben anhand von echten Beispieldaten, Programmierarbeiten, Gruppenarbeiten, Software-Live-Präsentationen und seminaristischer Unterricht.

Prüfungsformen:

Mündliche Prüfung oder Klausur.

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene mündliche Prüfung bzw. Klausur.

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

PD Dr. Martin Eisenacher

Sonstige Informationen:

Weitere Informationen unter <http://www.bioinf.rub.de/msc/>

Bioinformatik in der Proteomik II					
Modul-Nr: 201911	Credits: 5 CP	Workload: 150 h	Semester: 1-4 Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: V: 2 SWS; Ü: 1 SWS	Selbststudium: 105 h h	Gruppen- größe: 3 bis 20 Studie- rende
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>VL+Ü Bioinformatik der Proteomik I im Wintersemester (empfohlen, aber keine Voraussetzung), Grundlegende Programmierkenntnisse, Statistik-Grundkenntnisse</p>					
<p>Lernziele (learning outcomes):</p> <p>Fachspezifische Kompetenzen:</p> <p>Nach dem erfolgreichen Abschluss dieses Moduls</p> <ul style="list-style-type: none"> • haben die Studierenden die wichtigsten Sachverhalte aus der Vorlesung „Bioinformatik der Proteomik I“ als kurze Zusammenfassung kennengelernt (ein Vorlesungstermin), • sie lernen die Prinzipien von fortgeschrittenen Methoden der Bioinformatik der Proteomik kennen, • sie können diese weiterführenden Methoden, die aktuell zur Analyse von Rohdaten (Massenspektren) und Ergebnissen (Peptid-/Protein-Identifizierungen und -Quantifizierungen), sowie zu ihrer biologischen Interpretation, verwendet werden, erläutern und anwenden, • sie verstehen die diesen Methoden zugrundeliegenden algorithmischen und statistischen Ideen, • sie können Proteomik-spezifische Software und die Workflow-Engine KNIME nutzen, • sie können eigene Lösungsideen entwerfen und programmieren • und sie können die besprochenen Software-Tools und Methoden auf reale Daten und Fragestellungen anwenden. <p>Fachübergreifende/generische Kompetenzen:</p> <ul style="list-style-type: none"> • instrumentale Kompetenzen: - Intensive Nutzung der Lernplattform Moodle • systemische Kompetenzen: - Selbständiges Lernen und Arbeiten - Teamarbeit und Teamfähigkeit 					

KAPITEL 8. BIOINFORMATIK

- kommunikative Kompetenzen: - Präsentation der eigenen Arbeit und Ergebnisse - Vermittlung der in der Bioinformatik angewandten englischen Fachbegriffe - Rhetorik und sprachliche Kompetenz (deutsch / englisch)

Inhalt:

- De-novo Peptid-Sequenzierung
- Proteinlistenvergleich
- Open Searches
- Dark Matter of Proteomics
- SRM, MRM und PRM
- Data Independent Acquisition (DIA)
- Statistik des Vergleichs von experimentellen Gruppen
- Cross-Omics-Analysen
- Proteinexpressions-Analysen (überwachte und unüberwachte Methoden)
- Enrichment-Analysen
- Netzwerkanalysen
- Aktuelle Entwicklungen
- Software-Tools im Bereich Bioinformatik der Proteomik (Übung)
- Praktische (Programmier-)Aufgaben (Übung)
- Workflow-Engine KNIME (Übung)

Lehrformen:

Vorlesung: Folien-basierte Vorlesung. Übung: Lösung von kleinen praktischen Problemen als Hausaufgaben anhand von echten Beispieldaten, Programmierarbeiten, Gruppenarbeiten, Software-Live-Präsentationen und seminaristischer Unterricht.

Prüfungsformen:

Mündliche Prüfung oder Klausur.

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene mündliche Prüfung bzw. Klausur.

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

PD Dr. Martin Eisenacher

Sonstige Informationen:

KAPITEL 8. BIOINFORMATIK

Weitere Informationen unter <http://www.bioinf.rub.de/msc/>

Bioimage Informatics					
Modul-Nr: 190800	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vorlesung (+Übung)			Kontaktzeit: 4 SWS	Selbststudium: 90 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse: Grundlegende Programmierkenntnisse</p>					
<p>Lernziele (learning outcomes): Die Studierenden sollen elementare und aktuelle Techniken der Bildverarbeitung kennenlernen, die zur Analyse mikroskopischer Bilddaten verwendet werden. Sie sollen die dahinterliegenden algorithmischen, mathematischen und statistischen Ideen verstehen, und lernen in der Praxis, wie diese Methoden auf reale Daten und Fragestellungen angewendet werden.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Algorithmetik • Programmieren 					
<p>Inhalt: Die Analyse von mikroskopischen Bilddaten mit Methoden der Bildverarbeitung ist in den vergangenen Jahren ein wichtiges Thema in vielen Anwendungen in den Lebenswissenschaften geworden. In der Vorlesung werden grundlegende Konzepte der Verarbeitung mikroskopischer Bilddaten und deren Anwendungen behandelt. Die Vorlesung gliedert sich wie folgt:</p> <ul style="list-style-type: none"> • Grundlagen der Mikroskopie: Durchlicht- und Konfokalmikroskopie; Fluoreszenzmikroskopie; Mikroskopie in der Histopathologie; Mikroskopie in der Zellbiologie; Mikroskopie in der Neurobiologie • Morphologische Bildanalyse • Texturelle Bildanalyse • Kolokalisations-Verfahren • Algorithmen zur Bildregistrierung 					

KAPITEL 8. BIOINFORMATIK

- Algorithmen zum Verfolgen von Bewegungsmustern von Zellen (“Cell Tracking”)
- Verfahren zur Rekonstruktion von Neuronen (“Neuron Tracing”)
- Deep Learning Methoden zur mikroskopischen Bilddatenanalyse
- Methoden der Analyse von markerfreien Mikroskopie-Daten
- Methoden zur überwachten und unüberwachten Segmentierung von mikroskopischen Bildern

Lehrformen:

Vorlesung (+Übung)

Prüfungsformen:

mündliche Prüfung

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

PD Dr. Axel Mosig

Sonstige Informationen:

Weitere Informationen unter <http://www.bioinf.rub.de/msc/>

Master-Praktikum Big Data in der Bioinformatik					
Modul-Nr: 202621	Credits: 5 CP	Workload: 150 h	Semester: 1-4 Sem.	Turnus: immer	Dauer: 1 Semester
Lehrveranstaltungen: a) Praktikum			Kontaktzeit: 4 SWS	Selbststudium: 90 h	Gruppen- größe: 1-3 Studierende
<p>Teilnahmevoraussetzungen: praktische Programmiererfahrung in einer oder mehreren Sprachen (z. B. Java, R, Python, Matlab)</p> <p>Empfohlene Vorkenntnisse: Vertiefungsmodul(e) aus dem Themengebiet der Bioinformatik</p>					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls - können Studierende praktischen Herausforderungen bei der Entwicklung von Bioinformatik-Anwendungen begegnen, - haben die Studierenden Programmier-Bibliotheken aus dem Bereich Bioinformatik kennen gelernt - haben Studierenden die Verwendung von Workflow-Systemen eingeübt - haben Studierende die Mechanismen von Code- und Programm-Dokumentation eingeübt - haben Studierende Verfahren zur Bereitstellung eigener Libraries (z. B. R-Pakete, .jar-Files) angewendet</p> <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Literaturrecherche und Dokumentation • Projekt- und Zeitmanagement • Präsentation wissenschaftlicher Ergebnisse • Rhetorik und sprachliche Kompetenz • Selbständiges Lernen und Arbeiten • Teamarbeit und Teamfähigkeit <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Programmieren 					
<p>Inhalt: Die Bioinformatik wendet naturgemäß Informatik-Methoden auf die Daten eines lebenswissenschaftlichen Anwendungsfaches an, stellt also per se angewandte Digitalisierung dar. Das Praktikum vermittelt Grundlagen der Programmierung mit Bezug zu lebenswissenschaftlichen Anwendungen mit großen Datenmengen. Dies geschieht anhand aktueller Beispiele aus den drei Themengebieten Sequenzanalyse, Bildverarbeitung und Bioinformatik der Proteomik. Nach einer kurzen Einführung (Präsenztreffen) in Programmierung und Entwicklungs-</p>					

KAPITEL 8. BIOINFORMATIK

umgebungen (z. B. Java, R, C++, eclipse, RStudio, Python, Matlab) werden praktische Programmieraufgaben ausgegeben und im Laufe des Semesters bearbeitet, ggfls. mit weiteren Präsenztreffen zur Diskussion des Fortschritts.

Lehrformen:

Einführung als seminaristischer Unterricht, Bearbeitung der praktischen Aufgabe selbständig oder als Gruppenarbeit

Prüfungsformen:

Protokoll max. 10 Seiten (wenn Abgabe bis zwei Wochen nach Praktikumsende, dann Korrekturrunde vor Benotung), Dokumentation der praktischen Aufgabe

Voraussetzungen für die Vergabe von Kreditpunkten:

Protokoll (siehe Prüfungsformen), Bearbeitung / Dokumentation der praktischen Aufgabe, Teilnahme an den Präsenztreffen, Abschlusspräsentation (15 min. + 5 min. Diskussion)

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:**PD Dr. Martin Eisenacher**

PD Dr. Martin Eisenacher, Prof. Dr. Axel Mosig, Prof. Dr. Sven Rahmann (Universität Duisburg/Essen und TU Dortmund)

Sonstige Informationen:

Weitere Informationen unter <http://www.bioinf.rub.de/msc/>

Algorithmische Bioinformatik					
Modul-Nr: 042611	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: nach Ankündi- gung, ca. alle 3 Semester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: 20
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Abiturwissen Biologie und praktische Programmiererfahrung in mehreren Sprachen, etwa C, R, Python oder andere.</p>					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls:</p> <ul style="list-style-type: none"> • kennen und verstehen die Studierenden klassische und aktuelle Algorithmen und probabilistische Modelle der Bioinformatik (fachspezifische Kompetenz) • können die Studierenden die behandelten Algorithmen und Modelle schrittweise herleiten und erklären (kommunikative Kompetenz) • sind die Studierenden in der Lage, anwendungsbezogen angemessene Methoden und Modelle auszuwählen (systemische Kompetenz) • können die Studierenden Vor- und Nachteile einzelner Techniken beurteilen (systemische Kompetenz) • sind die Studierenden in der Lage, neue Varianten der behandelten Methoden zu entwerfen (systemische Kompetenz) und zu erklären (kommunikative Kompetenz) Die Erreichung der Lernziele setzt eine intensive Beschäftigung mit digitalen Technologien voraus, da es zu jeder in der Vorlesung behandelten Methode quelloffenen Code gibt, der sich studieren lässt. 					
<p>Inhalt:</p> <p>Diese Vorlesung behandelt algorithmische Techniken und Probleme, die in der Molekularbiologie auftreten. Die behandelten Themen beinhalten z.B. Mustererkennung in Genomsequenzen, Verarbeitung von sequenzierten Genomen, phylogenetische Bäume und Vorhersagemethoden der Sekundärstruktur von RNA. Dabei spielen zentrale Techniken der Algorithmik eine wichtige Rolle, wie z.B. String- und Graphalgorithmen, Greedy Algorithmen, Dynamische Programmierung sowie Divide-and-Conquer Algorithmen. In den Übungen wird auf mittleren bis großen digital vorliegenden Datensätzen gearbeitet. Zur Ergänzung des Vorlesungsmaterials wird u.a. englische Originalliteratur bereitgestellt (Internationalisierung).</p>					

<p>Lehrformen:</p> <ul style="list-style-type: none">a) Vorlesung im Präsenzstudium (unterstützt durch Skript zum Selbststudium),b) Einzel- und Gruppenübungen im Präsenzstudium
<p>Prüfungsformen:</p> <p>mündliche Modulabschlussprüfung von 30 min nach Terminvereinbarung zu den Inhalten und vermittelten Kompetenzen der Vorlesung und der Übungen</p>
<p>Voraussetzungen für die Vergabe von Kreditpunkten:</p> <p>Bestandene Modulabschlussklausur</p>
<p>Verwendung des Moduls (in anderen Studiengängen):</p> <p>Dies ist Modul INF-MSc-606 in den Bachelorstudiengängen Informatik und Angewandte Informatik der TU Dortmund.</p>
<p>Stellenwert der Note für die Endnote: 6 / 112</p>
<p>Modulbeauftragte/r und hauptamtlich Lehrende:</p> <p>Prof. Dr. Sven Rahmann (Universität Duisburg-Essen) Prof. Dr. Sven Rahmann</p>

Computaional Omics					
Modul-Nr: 042525	Credits: 6 CP	Workload: 180 h	Semester: beliebiges Sem.	Turnus: nach Ankündigung, ca. alle 3 Semester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) praktische Übung			Kontaktzeit: 4 SWS	Selbststudium: 120 h	Gruppen- größe: 20
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Abiturwissen Biologie, Grundkenntnisse in Statistik und maschinellem Lernen, praktische Programmiererfahrung in mehreren Sprachen (C, R, Python, etc.)</p>					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls:</p> <ul style="list-style-type: none"> • kennen und verstehen die Studierenden aktuelle Methoden für die Datenanalyse auf großen Datensätzen aus den Lebenswissenschaften (fachspezifische Kompetenz) • können die Studierenden die genannten Methoden in Form von existierender Software einsetzen (instrumentale Kompetenz) • können die Studierenden die behandelten Methoden schrittweise herleiten und erklären (kommunikative Kompetenz) • sind die Studierenden in der Lage, anwendungsbezogen angemessene Methoden und Modelle auszuwählen (systemische Kompetenz) • können die Studierenden Vor- und Nachteile einzelner Methoden beurteilen (systemische Kompetenz) • sind die Studierenden in der Lage, neue Varianten der behandelten Methoden zu entwerfen (systemische Kompetenz) und zu erklären (kommunikative Kompetenz) Die Erreichung der Lernziele setzt eine intensive Beschäftigung mit digitalen Technologien voraus, da es zu jeder in der Vorlesung behandelten Methode quelloffenen Code gibt, der in den Übungen genutzt wird. 					
<p>Inhalt:</p> <p>Die Vorlesung gibt eine Übersicht über die aktuellen informatischen Methoden zur Datenanalyse in den “Omiken” der Lebenswissenschaften (Genomik, Transkriptomik, Epigenomik, Proteomik, Metabolomik, Interaktomik). Sie besteht aus mehreren Einheiten, die sich jeweils einem dieser Themenbereiche und zugehörigen Technologien widmen. Dabei liegt der Fokus auf den theoretischen Grundlagen und allgemeinen algorithmischen und statistischen Prinzi-</p>					

pien. Typische Vorlesungsthemen sind:

- Hochdurchsatz-DNA-Sequenzierung und Fehlerkorrektur
- Genomassemblierung
- Variantendetektion in Genomen
- Genexpressionsanalysen (Microarrays, RNA-seq)
- Massenspektrometrie: Identifikation eines Moleküls aus Masseninformationen
- Proteinidentifikation aus Massenfingerprints
- Ionenmobilitätspektrometrie in der Metabolomik

In den Einheiten wird jeweils eine Einführung in die zugrundeliegenden Technologien gegeben; dabei wird der Art und Erzeugung der Daten besondere Aufmerksamkeit gewidmet. Es schließen sich typische Fragestellungen an, die aktuell anhand der gewonnenen Daten gestellt und beantwortet werden können. Dazu werden jeweils die wichtigsten Datenanalysemethoden besprochen. Diese unterteilen sich häufig in sogenannte low-level-Verfahren zur Vorverarbeitung, die sich vor allem nach der Art der Daten richten und high-level-Verfahren aus dem Bereich des maschinellen Lernens, die die gewünschten Informationen aus den Daten extrahieren. Aufgrund des hierbei auftretenden Datenvolumens stehen dabei besonders ressourceneffiziente Algorithmen im Vordergrund. Aus statistischer Sicht geht es zusätzlich darum, sinnvoll mit dem Problem hochdimensionaler Daten bei kleiner Stichprobengröße umzugehen. In den (praktisch ausgerichteten) Übungen werden aktuelle Arbeiten zu Technologien und Algorithmen in Form von Miniprojekten vertieft und ausgearbeitet. In den Übungen wird auf mittleren bis großen digital vorliegenden Datensätzen gearbeitet. Zur Ergänzung des Vorlesungsmaterials wird u.a. englische Originalliteratur bereitgestellt (Internationalisierung).

Lehrformen:

- a) Vorlesung im Präsenzstudium (unterstützt durch Skript zum Selbststudium),
- b) Einzel- und Gruppenübungen im Präsenzstudium

Prüfungsformen:

mündliche Modulabschlussprüfung von 30 min nach Terminvereinbarung zu den Inhalten und vermittelten Kompetenzen der Vorlesung und der Übungen

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Verwendung des Moduls (in anderen Studiengängen):

Dies ist Modul INF-MSc-514 in den Masterstudiengängen Informatik und Angewandte Informatik der TU Dortmund.

Stellenwert der Note für die Endnote: 6 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Sven Rahmann (Universität Duisburg-Essen)

Prof. Dr. Sven Rahmann

Sonstige Informationen:

Die Veranstaltung verlangt eine vertiefende Beschäftigung mit fortgeschrittenen statistischen Fragestellungen und sollte daher nur von Studierenden belegt werden, die keine Berührungspunkte mit Mathematik haben.

Algorithmen auf Sequenzen					
Modul-Nr: 040329 / 040330	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: nach Ankündigung, ca. alle 3 Semester	Dauer: 1 Semester
Lehrveranstaltungen: a) Vorlesung b) Übung			Kontaktzeit: 3 SWS	Selbststudium: 105 h	Gruppengröße: 20
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>Kenntnisse über elementare Datenstrukturen und Algorithmen und Programmierfähigkeiten, wie sie üblicherweise in Einführungsveranstaltungen gelehrt werden, sowie Grundkenntnisse der theoretischen Informatik (Automatentheorie, reguläre Sprachen) und der Wahrscheinlichkeitsrechnung (Erwartungswerte).</p>					
<p>Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls:</p> <ul style="list-style-type: none"> • kennen und verstehen die Studierenden die die zu Grunde liegenden algorithmischen und mathematischen Konzepte hinter verschiedenen String-Algorithmen (fachspezifische Kompetenz) • können die Studierenden die genannten Methoden in Form von existierender Software einsetzen (instrumentale Kompetenz) • können die Studierenden die behandelten Methoden schrittweise herleiten und erklären (kommunikative Kompetenz) • sind die Studierenden in der Lage, anwendungsbezogen angemessene Methoden und Modelle auszuwählen (systemische Kompetenz) • können die Studierenden Vor- und Nachteile einzelner Methoden beurteilen (systemische Kompetenz) • sind die Studierenden in der Lage, neue Varianten der behandelten Methoden zu entwerfen (systemische Kompetenz) und zu erklären (kommunikative Kompetenz) Die Erreichung der Lernziele setzt eine intensive Beschäftigung mit digitalen Technologien voraus, da es zu jeder in der Vorlesung behandelten Methode quelloffenen Code gibt, der in den Übungen genutzt wird. 					
<p>Inhalt:</p> <p>Die Vorlesung gibt einen weiträumigen Überblick über das Problem der Mustersuche in Texten. Besprochen werden verschiedene Arten von Mustern: einfache Strings, Mengen von</p>					

Strings, verallgemeinerte Strings, eingeschränkte reguläre Ausdrücke, Muster mit wiederholten und optionalen Zeichen, allgemeine reguläre Ausdrücke und positionsspezifische Gewichtsmatrizen. Verschiedene algorithmische Prinzipien werden vorgestellt: endliche Automaten, Bit-Parallelität, Orakel. Um Strings zu vergleichen, werden unterschiedliche Distanz- und Ähnlichkeitsmaße zwischen Sequenzen definiert und sogenannte Gapkosten-Modelle eingeführt. Die folgenden algorithmische Techniken für die approximative Mustersuche werden eingeführt: Automaten, Bit-Parallelität, dynamische Programmierung, Four-Russians-Trick und Tabellierung. Verschiedene Volltext-Index-Datenstrukturen werden vorgestellt: q-Gramm-Index, Suffixbaum, Suffixarray, FM-Index und Burrows-Wheeler-Transformation. Effiziente Konstruktionsalgorithmen für diese Index-Datenstrukturen werden eingeführt, z.B. Ukkonen's Algorithmus und das "Induced Sorting" (SA-IS Algorithmus). Verschiedene Anwendungen, insbesondere das effiziente Finden wiederholter Teilstrings, werden besprochen, sowie Bezüge zu Kompressionsalgorithmen aufgezeigt. Sowohl in der biologischen Sequenzanalyse als auch bei der Revisions- und Versionsverwaltung von Dokumenten ist das Konzept des paarweisen Sequenzalignment zentral; verschiedene Alignment-Typen werden besprochen und Algorithmen dafür vorgestellt. Die Laufzeit aller vorgestellten Algorithmen wird analysiert; dabei kommt das Konzept der amortisierten Analyse häufig vor. Die Übungsaufgaben unterteilen sich in Beispiele, Verständnisfragen, Erweiterungen des Lehrstoffs, Implementierungsaufgaben und praxisbezogene Anwendungsaufgaben.

Lehrformen:

- a) Vorlesung im Präsenzstudium (unterstützt durch Skript zum Selbststudium),
- b) Einzel- und Gruppenübungen im Präsenzstudium

Prüfungsformen:

mündliche Modulabschlussprüfung von 25 min nach Terminvereinbarung zu den Inhalten und vermittelten Kompetenzen der Vorlesung und der Übungen

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Verwendung des Moduls (in anderen Studiengängen):

Dies ist Modul INF-BSc-315 in den Bachelorstudiengängen Informatik und Angewandte Informatik der TU Dortmund.

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Sven Rahmann (Universität Duisburg-Essen)

Prof. Dr. Sven Rahmann

Sonstige Informationen:

Im Bachelorstudiengängen Informatik und Angewandte TU Dortmund gibt es für dieses Modul 4 CP. Bioinformatik-Studierende der RUB bearbeiten umfangreichere Aufgaben, so dass sich die Zeit im Selbststudium auf 105 h und sich damit der Modulumfang auf 5 CP erhöht.

Masterseminare

Seminar Ingenieurinformatik					
Modul-Nr: 125017	Credits: 3 CP	Workload: 90 h	Semester: 2-3 Sem.	Turnus: Winter- und Sommersemes- ter	Dauer: 1 Semester
Lehrveranstaltungen: Seminar Ingenieurinformatik			Kontaktzeit: 2 SWS	Selbststudium: 60 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse:</p> <p>Mindestens ein Vertiefungsmodul der Ingenieurinformatik</p>					
<p>Lernziele (learning outcomes):</p> <p>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"> • kennen Studierende die methodischen Ansätze und Techniken, um sich in eine Thematik gründlich einzuarbeiten. Dabei sollen die gewonnenen Erkenntnisse in verständlicher Form weitergegeben werden; • verstehen Studierende durch Bearbeitung einer Seminararbeit, wie die erarbeiteten Fakten, Informationen und Zusammenhänge in Rahmen einer Präsentation klar vermittelt werden können. <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten 					

KAPITEL 9. MASTERSEMINARE

- Kritikfähigkeit
- Literaturrecherche und Dokumentation
- Projekt- und Zeitmanagement
- Präsentation wissenschaftlicher Ergebnisse
- 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 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Markus König
Dr. K. Lehner

Seminar Kryptologie und Theoretische Informatik					
Modul-Nr: 150537: Se- minar zur Kryptographie	Credits: 3 CP	Workload: 90 h	Semester: beliebiges Sem.	Turnus: Unregelmäßig	Dauer: 1 Semester
Lehrveranstaltungen: Vertiefungsseminare			Kontaktzeit: 2 SWS	Selbststudium: 60 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Keine					
Empfohlene Vorkenntnisse: Kryptographie, Kryptanalyse, Algorithmik					
Lernziele (learning outcomes): Es werden aktuelle Forschungsarbeiten der Kryptologie und Algorithmik vorgestellt					
Vermittelte Kompetenzen: Kernkompetenzen:					
<ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Kritikfähigkeit • Literaturrecherche und Dokumentation • Projekt- und Zeitmanagement • Präsentation wissenschaftlicher Ergebnisse • Rhetorik und sprachliche Kompetenz • Selbständiges Lernen und Arbeiten 					
Inhalt: Inhalt entsprechend des gewählten Seminars.					
Lehrformen: Vertiefungsseminare					
Prüfungsformen: Seminar-Arbeit					
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur					
Stellenwert der Note für die Endnote: 3 / 112					
Modulbeauftragte/r und hauptamtlich Lehrende:					

KAPITEL 9. MASTERSEMINARE

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

Seminar Computerlinguistik					
Modul-Nr: 050025: Tools für den linguistischen Alltag: Automatisierung, Reproduzierbarkeit und Kollaboration; 050042: Dialogsysteme; 050042: Koreferenzauflösung	Credits: 3 CP	Workload: 90 h	Semester: beliebiges Sem.	Turnus: Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vertiefungsseminare			Kontaktzeit: 2 SWS	Selbststudium: 60 h	Gruppengröße: beliebig
Teilnahmevoraussetzungen: Linguistische Grundlagen, Linguistische Schwerpunkte und Linguistische Methoden					
Lernziele (learning outcomes): <ol style="list-style-type: none"> 1. Fortgeschrittene Kenntnisse in unterschiedlichen Bereichen der Computerlinguistik und Sprachtechnologie 2. Fähigkeit, die für computerlinguistische Themen und Aufgabenstellungen jeweils geeigneten Analysemethoden sowie die entsprechenden Algorithmen und Implementationsverfahren auswählen, bewerten, und anwenden zu können 3. generelle Kompetenz in eigenständiger computerlinguistischer Analyse, Modellierung und Implementierung sprachlicher Strukturen und Prozesse 					
Vermittelte Kompetenzen: Kernkompetenzen: <ul style="list-style-type: none"> • 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 / 112

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.

Seminar Operations Research (Management Science)					
Modul-Nr: WS: 074230, SS: 074232	Credits: 5 CP	Workload: 150 h	Semester: beliebiges Sem.	Turnus: Letztmalig im SS 20!	Dauer: 1 Semester
Lehrveranstaltungen: Vertiefungsseminare			Kontaktzeit: WS: 4, SS: 6 SWS	Selbststudium: 150 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Fundierte Kenntnisse der Anwendung quantitativer Methoden aus dem Bachelorstudium oder der Mastermodule "Rationales Entscheiden" und "Management Science" werden vorausgesetzt.</p> <p>Empfohlene Vorkenntnisse:</p> <p>Betriebswirtschaftliche und entscheidungstheoretische Kenntnisse sowie Programmierkenntnisse sind hilfreich.</p>					
<p>Lernziele (learning outcomes): Studierende erwerben vertiefte Kenntnisse quantitativer Methoden und Anwendungen zur optimalen Lösung vielfältiger wirtschaftlicher Fragestellungen, die teils von Praxispartner stammen. Gleichzeitig wird die Fähigkeit zur selbständigen Auseinandersetzung mit komplexen Problemen auf wissenschaftlichem Niveau und unter Berücksichtigung des aktuellen Stands der Forschung erreicht.</p> <p>Vermittelte Kompetenzen:</p> <p>Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten • Selbständiges Lernen und Arbeiten • Kritikfähigkeit • Projekt- und Zeitmanagement • Rhetorik und sprachliche Kompetenz • Literaturrecherche und Dokumentation • Präsentation wissenschaftlicher Ergebnisse • Wissenschaftliches Arbeiten und Schreiben <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Optimierungsmethoden 					

KAPITEL 9. MASTERSEMINARE

- Simulationsmethoden
- Wirtschaftliches Handeln

Inhalt:

Jeweils ausgewählte Gebiete des Management Science werden vertiefend behandelt. Dazu werden Themenstellungen durch die Teilnehmer selbständig bearbeitet und in Form einer Hausarbeit und einer Präsentation zur Diskussion gestellt. Damit erfolgt auch eine Vorbereitung auf die Masterarbeit, insbesondere im Bereich Operations and Service Management.

Die Modulnote setzt sich aus der Note der Hausarbeit (70 %) und der für Präsentation und Diskussion im Plenum (30 %) zusammen.

Seminare:

Angeboten über die Fakultät für Wirtschaftswissenschaft Vorlesungsnr.: 074231 (Sommersemester 2013) Vorlesungsnr.: 074232 (Wintersemester 2013/14) Modul Seminar Management Science geplantes Angebot jeweils im Sommer- und Wintersemester

Lehrformen:

Vertiefungsseminare

Prüfungsformen:

Seminar-Arbeit

Voraussetzungen für die Vergabe von Kreditpunkten:

Bestandene Modulabschlussklausur

Stellenwert der Note für die Endnote: 5 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Brigitte Werners

Sonstige Informationen:

Letztmalig im SS 20!

Literatur:

Veranstaltungsunterlagen und Literaturhinweise aus wissenschaftlichen Zeitschriften werden zu Beginn der Veranstaltung angekündigt bzw. bereitgestellt.

Seminar Computational Cognitive Modeling					
Modul-Nr: 310024	Credits: 3 CP	Workload: 60 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Vertiefungsseminare			Kontaktzeit: 2 SWS	Selbststudium: 30 h	Gruppen- größe: beliebig
Teilnahmevoraussetzungen: Basic knowledge of cognition, e.g. "Cognition I + II", "Learning"					
<p>Lernziele (learning outcomes): MISSING</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Analytisches und logisches Denken • Fächerübergreifendes Denken und Arbeiten • Kritikfähigkeit • Präsentation wissenschaftlicher Ergebnisse • Rhetorik und sprachliche Kompetenz • Selbständiges Lernen und Arbeiten <p>Fachspezifische Kompetenzen:</p> <ul style="list-style-type: none"> • Künstliche Intelligenz und Maschinelles Lernen 					
<p>Inhalt: The human mind is most intimately familiar to us, yet we understand very little about how it functions. To study the mind, the field of cognitive science pursues an interdisciplinary approach. One of the pillars of cognitive science is computational modeling. This seminar will survey models of perception, memory and action. Rather than focusing on the mathematical details, we will discuss the motivation, application and noteworthy properties of the models, including their strengths and shortcomings. Class work will include student presentations and discussions.</p>					
Lehrformen: Vertiefungsseminare					
Prüfungsformen: presentation in class					
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur					

Stellenwert der Note für die Endnote: 3 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Sen Cheng
Sonstige Informationen: Course Material will be available in Blackboard (sign-up required). Literatur: "The Cambridge Handbook of Computational Psychology" edited by Ron Sun, Cambridge University Press + modeling papers announced in class

Seminar Bioinformatik					
Modul-Nr: 190559	Credits: 3 CP	Workload: 90 h	Semester: beliebiges Sem.	Turnus: Wintersemester	Dauer: 1 Semester
Lehrveranstaltungen: Praktikum			Kontaktzeit: 2 SWS	Selbststudium: 60 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: keine</p> <p>Empfohlene Vorkenntnisse: Mindestens ein Vertiefungsmodul auf dem Themengebiet der Bioinformatik</p>					
<p>Lernziele (learning outcomes): Lernziel ist die selbständige Auseinandersetzung mit einem Thema aus dem Bereich der Bioinformatik. Es soll die Fähigkeit der kritischen Auseinandersetzung mit einem Thema im Rahmen einer Fachdiskussion gefördert werden.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten • Kritikfähigkeit • Literaturrecherche und Dokumentation • Projekt- und Zeitmanagement • Präsentation wissenschaftlicher Ergebnisse • Rhetorik und sprachliche Kompetenz • Selbständiges Lernen und Arbeiten 					
<p>Inhalt: Die Themen werden zu Semesterbeginn bekanntgegeben und werden jeweils aus aktuellen Themen der Bioinformatik zusammengestellt, insbesondere aus den Themengebieten</p> <ul style="list-style-type: none"> • Bioinformatik der Proteomik • Bioinformatik der Sequenzanalyse • Analyse mikroskopischer Bilddaten • Analyse von Omics Daten • Bioinformatik und molekulare Evolution 					

KAPITEL 9. MASTERSEMINARE

Lehrformen: Praktikum
Prüfungsformen: Seminararbeit
Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur
Stellenwert der Note für die Endnote: 3 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: PD Dr. Axel Mosig PD Dr. Martin Eisenacher; Prof. Dr. Axel Mosig; Prof. Dr. Sven Rahmann

Seminar Algorithmen					
Modul-Nr: 125017	Credits: 5 CP	Workload: 90 h	Semester: 1-3 Sem.	Turnus: Jedes zweite Semester (SoSe)	Dauer: 1 Semester
Lehrveranstaltungen: Seminar			Kontaktzeit: 2 SWS	Selbststudium: 120 h	Gruppen- größe: 12
Teilnahmevoraussetzungen: Keine					
Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • Können Studierende Literatur zu einem algorithmischem Thema sich selber erarbeiten • Können Studierende ein algorithmisches Thema mündlich vortragen • Haben Studierende andere Vorträge gesehen und können zu diesen Fragen stellen 					
Inhalt: In dem Seminar wird ein aktuelles Thema der Algorithmik besprochen. Dies können zum Beispiel parametrisierte oder verteilte Algorithmen sein, oder Algorithmen mit Anwendungen in der Robotik oder in Geographischen Informationssystemen.					
Lehrformen: Seminarvorträge der Studierenden mit anschließender Diskussion					
Prüfungsformen: Vortrag von 45-60 min, Ausarbeitung zum Vortrag von max. 10 Seiten					
Voraussetzungen für die Vergabe von Kreditpunkten: Erfolgreicher Vortrag und Ausarbeitung					
Verwendung des Moduls (in anderen Studiengängen): Informatik, Mathematik					
Stellenwert der Note für die Endnote: 5 / 112					
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Maike Buchin					

Studienprojekt

Master-Studienprojekt					
Modul-Nr: N/A	Credits: 10 CP	Workload: 300 h	Semester: beliebiges Sem.	Turnus: immer	Dauer: 1 Semester
Lehrveranstaltungen: Praktikum/Projektarbeit			Kontaktzeit: 2 SWS	Selbststudium: 270 h	Gruppen- größe: beliebig
<p>Teilnahmevoraussetzungen: Keine Empfohlene Vorkenntnisse:</p> <p>Bitte entnehmen Sie die projektspezifischen Vorkenntnisse dem jeweiligen Studienprojekt.</p>					
<p>Lernziele (learning outcomes): Ziele des Studienprojekts sind:</p> <ul style="list-style-type: none"> • Die Anwendung des erlernten Fachwissens. • Der Erwerb zusätzlicher Fachkompetenz gemäß der jeweiligen projektspezifischen Aufgabenstellung. • Die Schulung hinsichtlich der Erarbeitung eigener Lösungsstrategien. • Die Schulung hinsichtlich Arbeitsteilung und Zusammenarbeit im Team (Teamfähigkeit und Projektorganisation). • Erwerb weiterer Kompetenz im Hinblick auf die Dokumentation und die Präsentation von Ergebnissen. <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Kritikfähigkeit • Projekt- und Zeitmanagement • Präsentation wissenschaftlicher Ergebnisse 					

KAPITEL 10. STUDIENPROJEKT

- Teamarbeit und Teamfähigkeit
- Fächerübergreifendes Denken und Arbeiten

Inhalt:

Im Rahmen des Studienprojekts soll eine Aufgabe aus Bereichen der Angewandten Informatik in Teamarbeit unter Anleitung eines Betreuers gelöst werden.

Organisation und Ablauf: Das Studienprojekt muss vor Beginn beim Prüfungsamt angemeldet werden. Bitte beachten Sie, die hierfür festgelegten Fristen unter Sonstiges. 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: 10 / 112

Modulbeauftragte/r und hauptamtlich Lehrende:

Prof. Dr. Thomas Herrmann
Dozenten der RUB

Freie Wahlfächer

Freie Wahlfächer					
Modul-Nr: N/A	Credits: Mindestens 8 CP	Workload: 240 h	Semester: beliebiges Sem.	Turnus: Winter- und Sommersemester	Dauer: 1 Semester
Lehrveranstaltungen: Frei Wählbar			Kontaktzeit: Je nach Lehrveranstaltung SWS	Selbststudium: 240 h	Gruppengröße: beliebig
<p>Teilnahmevoraussetzungen: Keine</p> <p>Empfohlene Vorkenntnisse: Abhängig von der gewählten Veranstaltung</p>					
<p>Lernziele (learning outcomes): Die Teilnehmer erwerben so genannte Schlüsselfähigkeiten in den freien Wahlfächern.</p> <p>Vermittelte Kompetenzen: Kernkompetenzen:</p> <ul style="list-style-type: none"> • Fächerübergreifendes Denken und Arbeiten 					
<p>Inhalt: Die freien 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>Lehrformen: Frei Wählbar</p>					
<p>Prüfungsformen: Abhängig von der gewählten Veranstaltung</p>					
<p>Voraussetzungen für die Vergabe von Kreditpunkten: Bestandene Modulabschlussklausur</p>					

Stellenwert der Note für die Endnote: Mindestens 8 / 112
Modulbeauftragte/r und hauptamtlich Lehrende: Prof. Dr. Laurenz Wiskott Dozenten der RUB
Sonstige Informationen: Aus der Fakultät für Wirtschaftswissenschaften sind nur die folgenden Veranstaltungen als freie Wahlfächer für die AI geöffnet: <ul style="list-style-type: none">• Finanzierung und Investition• Grundlagen der Businessplanerstellung• Grundlagen der Existenzgründung• Grundlagen der Makroökonomie• Grundlagen der Mikroökonomie• Grundlagen des Wirtschaftsrechts• Jahresabschluss• Märkte und Unternehmungen (sofern nicht bereits im Bachelor absolviert)• Marktorientierte Unternehmensführung (sofern nicht bereits im Bachelor absolviert) Der Besuch weiterer Veranstaltungen ist nur nach Absprache mit dem Dozenten und dem Prüfungsamt AI möglich.