

# Modulhandbuch für Software Systems Engineering (Master 1 Fach)



Prüfungsordnungsbereich



Modulangebot



Prüfungsangebot



Lehrangebot

Prüfungsordnungsbeschreibung:	7 >
Wahlpflichtbereiche:	8 >
Theoretical Foundations of Software Systems Engineering	8 >
Wahlpflichtbereich Theoretische Grundlagen des SSE	8 >
[1211981] Advanced Automata Theory	8 >
[1217537] Algorithmische Lerntheorie	10 >
[1113583] Algorithmische Modelltheorie I	12 >
[1212326] Algorithmische Spieltheorie	14 >
[1212337] Analyse von Algorithmen	16 >
[1211978] Compilerbau	18 >
[1212646] Concurrency Theory	20 >
[1226911] Fixpoints and Induction in Logic and Computer Science	22 >
[1212327] Graphalgorithmen	24 >
[1212710] Graphzerlegungen und algorithmische Anwendungen	26 >
[1223638] Höhere Algorithmik	28 >
[1212336] Infinite Computations and Games	30 >
[1221327] Introduction to Algorithmic Differentiation	32 >
[1230246] Introduction to Quantum Computing	34 >
[1212331] Komplexitätstheorie	36 >
[1113594] Komplexitätstheorie und Quantum Computing	38 >
[1113599] Logik und Spiele	40 >
[1112957] Mathematische Logik II	42 >
[1212711] Modeling and Verification of Probabilistic Systems	44 >
[1212339] Modellierung und Analyse hybrider Systeme	46 >
[1230345] Near-term Quantum Computation	48 >
[1212645] Online Algorithmen	50 >
[1212338] Parametrisierte Algorithmen	52 >
[1229347] Post-quantum cryptography	54 >
[1212650] Probabilistic Programming	56 >
[1114998] Quantum Computing	58 >
[1212335] Rekursionstheorie	60 >
[1212329] Semantik und Verifikation von Software	62 >
[1212330] Statische Programmanalyse	64 >
[1219623] Stochastic Games	66 >
[1212654] The Graph Isomorphism Problem	68 >
[1212334] Theory of Constraint Satisfaction Problems	70 >
Core Subjects Theoretical Foundations of SSE	72 >
[1216860] Algorithmic Foundations of Datasience	72 >
[1212341] Erfüllbarkeitsüberprüfung	74 >
[1212648] Formale Grundlagen von UML	76 >
[1212666] Formale Methoden für Steuerungssoftware	78 >
[1215684] Foundations of Functional Programming	80 >

[1212343] Foundations of Logic Programming.....	82 >
[1212328] Model Checking.....	84 >
[1212643] Theory of Distributed and Parallel Systems.....	86 >
<b>Communication.....</b>	<b>88 &gt;</b>
[6010396] Ad-Hoc Networks: Architectures and Protocols.....	88 >
[1215688] Advanced Internet Technology.....	90 >
[1230527] Advanced Network Security.....	92 >
[1212349] Communication Systems Engineering.....	94 >
[1223640] Dynamical Processes on Networks.....	96 >
[1227956] Industrial Network Security.....	98 >
[1211901] IT-Sicherheit 1 - Kryptographische Grundlagen und Netzwerksicherheit.....	100 >
[1212346] Mobile Internet Technology.....	102 >
[6010380] Mobile Radio Networks 1.....	104 >
[6010432] Mobile Radio Networks 2.....	106 >
[1212347] Research Focus Class on Communication Systems.....	108 >
<b>Data and Information Management.....</b>	<b>110 &gt;</b>
[1228568] Actions and Planning in AI: Learning, Models, and Algorithms.....	110 >
[1220136] Advanced Process Mining.....	112 >
[1216958] Business Process Intelligence.....	114 >
[1215691] CSCW and Groupware: Concepts and Systems for Computer Supported Cooperative Work.....	116 >
[1226146] Datenstrommanagement und -analyse.....	118 >
[1226006] Distributed Ledger Technology.....	121 >
[1212683] eBusiness - Anwendungen, Architekturen und Standards.....	124 >
[1227457] Fundamentals of Business Process Management.....	126 >
[1215692] Implementation of Databases.....	128 >
[1211903] Introduction to Bioinformatics.....	130 >
[1216861] Introduction to Data Science.....	132 >
[1211900] IT-Sicherheit 2 - Computer Security.....	134 >
[1215694] Künstliche Intelligenz.....	136 >
[1215751] Learning Technologies.....	138 >
[1212677] Privacy Enhancing Technologies for Data Science.....	141 >
[1211902] Prozess Management.....	143 >
[1222419] Research Focus Class on Learning Technologies.....	145 >
[1212675] Semantic Web.....	147 >
[1229157] Social and Technological Change.....	149 >
[1211393] The Logic of Knowledge Bases.....	151 >
[1222468] Uncertainty in Robotics.....	153 >
[1212361] Wissensrepräsentation.....	155 >
[1231482] Research Focus Class on Data Ecosystems.....	157 >
[1231539] Industrial Data Security.....	159 >
[1229308] Linux Kernel Programming.....	161 >

Applied Computer Science.....	163 >
[1221328] Advanced Algorithmic Differentiation.....	163 >
[1228566] Advanced C++.....	165 >
[1211912] Advanced Machine Learning.....	168 >
[1230105] Automatic Speech Recognition Search .....	170 >
[1215721] Combinatorial Problems in Scientific Computing.....	172 >
[1215724] Computer Vision.....	174 >
[1211921] Computer Vision 2.....	176 >
[1211908] Current Topics in Media Computing and HCI.....	178 >
[1215842] Data Driven Medicine - project-oriented, multidisciplinary introduction.....	180 >
[1215699] Designing Interactive Systems II.....	182 >
[1212688] Fortgeschrittene Methoden der Virtuellen Realität.....	184 >
[1215696] Geometrieverarbeitung.....	186 >
[1212310] Grundlagen der Computergraphik.....	188 >
[1215720] High-Performance Computing.....	190 >
[1221327] Introduction to Algorithmic Differentiation.....	192 >
[1229154] Introduction to Numerical Methods and Software with C++.....	194 >
[1215681] iOS Application Development.....	196 >
[1216838] Konzepte und Modelle der parallelen und datenzentrischen Programmierung.....	198 >
[1215744] Machine Learning.....	200 >
[1227996] Machine Learning with Graphs: Foundations and Applications.....	202 >
[9014711] Medical Image Processing.....	204 >
[1220524] Mikrocontrollerprogrammierung und Fehlersuche.....	206 >
[1215862] Physikalisch-Basierte Animation.....	208 >
[1212692] Physikalische Simulation im Visual Computing.....	210 >
[4026526] Reinforcement Learning and Learning-based Control.....	212 >
[1223639] Shape Analysis and 3D Deep Learning.....	214 >
[8015429] Simulation Software Engineering.....	216 >
[1215695] Statistische Methoden zur Verarbeitung natürlicher Sprache.....	218 >
[4029089] Stochastic Processes and Dynamical Systems.....	220 >
[1211909] Virtuelle Realität.....	222 >
[1230525] Advanced Topics in Machine Learning for Human Language Technology.....	224 >
[1211911] High-performance Matrix Computations.....	226 >
Software Engineering.....	228 >
Wahlpflichtbereich Software Engineering.....	228 >
[1215757] Angewandte Softwaretechnik im Lebenszyklus der Automobilelektronik.....	228 >
[1215755] Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT).....	230 >
[1220230] Digitalisierung.....	232 >
[1215690] Eingebettete Systeme.....	234 >
[1220524] Mikrocontrollerprogrammierung und Fehlersuche.....	236 >
[8015429] Simulation Software Engineering.....	238 >
[1212355] Software-Projektmanagement.....	240 >

	[1212356] Software-Qualitätssicherung.....	242 >
—	Core Subjects Software Engineering.....	244 >
+	[1215698] Designing Interactive Systems I.....	244 >
	[1212666] Formale Methoden für Steuerungssoftware.....	246 >
	[1215686] Modellbasierte Softwareentwicklung.....	248 >
	[1212354] Objektorientierte Softwarekonstruktion.....	250 >
	[1216957] Software Language Engineering.....	252 >
	[1215687] Software-Architekturen.....	254 >
	[1222882] Model-based Systems Engineering.....	256 >
—	Schwerpunktkolloquium.....	258 >
+	[1222540] Schwerpunktkolloquium.....	258 >
—	Seminar, Praktikum .....	260 >
+	[1215759] Praktikum.....	260 >
	[1211974] Seminar Informatik.....	262 >
—	Seminar aus anderen Fakultäten/Fachgruppen.....	264 >
+	[4028624] Seminar: Mathematical Concepts of Machine Learning.....	264 >
	[7023944] Current Topics in Applied Computational Social Sciences.....	266 >
—	Deutschkurs oder Ersatzveranstaltung.....	268 >
+	[1215734] Deutschkurs.....	268 >
—	Masterarbeit.....	270 >
+	[1212703] Masterarbeit.....	270 >

Prüfungsordnungsbeschreibung:  
Software Systems Engineering (SPO-Version / 2011)

Titel	Software Systems Engineering
Kurzbezeichnung	MSSSE
Version	2011
Studien- und Qualifikationsziele	<p>Verglichen mit Absolvent*innen des Bachelors haben Masterabsolvent*innen eine größere Reife und ein ausgeprägteres Verständnis, wenn es um die Anwendung oder den Transfer ihrer technischen und sozialen Kompetenzen geht.</p> <ul style="list-style-type: none"> <li>• Sie haben fundiertes technisches Wissen in ausgewählten spezialisierten Bereichen der Informatik gesammelt.</li> <li>• Ihre Kompetenzen sind sowohl ausreichend tief als auch breit gefächert, damit sie sich selbstständig mit zukünftigen Technologien in ihrer Disziplin und verwandten Gebieten vertraut machen können.</li> <li>• Sie sind befähigt Methoden der Informatik erfolgreich anzuwenden, kritisch zu hinterfragen und wenn nötig zu verbessern, um komplexe Probleme aus Industrie und Forschung zu lösen.</li> <li>• Sie haben sich verschiedene technische und soziale Kompetenzen angeeignet (z.B. die Fähigkeit zu abstrahieren, analytisches und systemorientiertes Denken, die Fähigkeit in Arbeitsgruppen zu arbeiten und zu kommunizieren, internationale und interkulturelle Erfahrungen), mit denen sie bevorzugt in führenden Positionen eingesetzt werden können.</li> <li>• Sie sind nicht nur für Fragestellungen in der Forschung und Entwicklung geeignet, sondern auch gut auf andere herausfordernde Aufgaben, wie Führungsaufgaben in der Wirtschaft und Verwaltung vorbereitet.</li> </ul> <p>Zugangsvoraussetzung für das Master-Programm ist ein abgeschlossenes Bachelorstudium in Informatik oder einer nah verwandten Fachrichtung. Bei Bewerber*innen mit anderen Abschlüssen werden individuell die Zulassungsvoraussetzungen geprüft. Absolvent*innen des Masterstudiengangs Software Systems Engineering der RWTH Aachen sind fähig, wissenschaftliche Methoden der Informatik anzuwenden, diese Methoden in den speziellen Bereichen weiterzuentwickeln und verantwortungsvoll hinsichtlich der Konsequenzen des technologischen Fortschritts zu handeln. Für besonders qualifizierte Studierende ist der Master zudem die wissenschaftliche Zulassungsvoraussetzung zu einem Promotionsprogramm. Um das generelle Wissen der Informatik zu vertiefen, wählt jede*r Studierende verschiedene Spezialisierungen innerhalb der Informatik und die damit verbundenen weiterführenden Kurse und Seminare. Abgeschlossen wird diese Spezialisierung in einer wissenschaftlichen Masterarbeit, welche sowohl umfangreicher als auch anspruchsvoller als eine Bachelorarbeit ist. Zusätzlich haben Studierende vertieftes Wissen im Bereich des Software Engineering, das Absolventinnen und Absolventen erlaubt, an großen Softwareprojekten teilzunehmen und solche Projekte zu managen.</p>
Qualifikationsprofil	
Weitere Informationen	

Modultitel	Advanced Automata Theory (Wahlpflichtfach)
Kennung	1211981
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• Das Minimierungsproblem für nichtdeterministische Automaten</li> <li>• Zusammenhang zwischen Automaten und Logik</li> <li>• Automaten auf endlichen Bäumen</li> <li>• Algorithmen für unendliche Transitionssysteme</li> <li>• Grundlegende Unentscheidbarkeitsergebnisse in der Automatentheorie</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse: Die Studierenden lernen erweiterte Automatenmodelle und deren Eigenschaften kennen.</p> <p>Fähigkeiten: Die Studierenden erwerben die Fähigkeit, Modelle nach ihren grundlegenden Eigenschaften im Bezug auf ihre Ausdrucksfähigkeit und algorithmische Komplexität einzuschätzen</p> <p>Kompetenzen: Die Studierenden beherrschen die grundlegenden Konzepte zustandsbasierter Modelle der Informatik.</p>
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Es werden Kenntnisse aus den Bereichen 'Formale Systeme, Automaten und Prozesse', 'Berechenbarkeit und Komplexität' sowie 'Mathematische Logik' erwartet.
Literatur	-
Sprache	Deutsch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Dr. rer. nat. Christof Löding
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Advanced Automata Theory (1211981)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Advanced Automata Theory (121198102)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Advanced Automata Theory (121198101)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Advanced Automata Theory	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Lerntheorie (1217537)

Modultitel	Algorithmische Lerntheorie (Wahlpflichtfach)
Kennung	1217537
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• PAC learning</li> <li>• uniform convergence and sample size bounds</li> <li>• VC dimension</li> <li>• theoretical analysis of machine learning algorithms</li> <li>• advanced theoretical topics in machine learning</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: Theoretical understanding of machine models and algorithms and their limitations</p> <p>Skills: Clear conception of the assumptions underlying machine learning algorithms Ability to give estimates on the generalisation error and the quality of hypotheses</p> <p>Competence: Application of theoretical insights to support the selection of suitable machine learning algorithms and models</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Gute Kenntnisse aus 'Lineare Algebra', 'Stochastik', 'Datenstrukturen und Algorithmen'. Kenntnisse grundlegender Konzepte des Machine Learning (z.B. aus den Kursen 'Foundations of Data Science' oder 'Machine Learning').
Literatur	Shalev-Schwartz and Ben David. Understanding Machine Learning. Cambridge University Press, 2014.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher Informatikmodellierungsteamverantwortlicher:</p> <p>Dr. rer. nat. Katja Petzoldt</p> <p>Modulverantwortlicher:</p> <p>Universitätsprofessor Dr. rer. nat. Martin Grohe</p>
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Lerntheorie (1217537)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Algorithmische Lerntheorie (121753702)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Algorithmische Lerntheorie (121753701)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Algorithmische Lerntheorie	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Modelltheorie I (1113583)

Modultitel	Algorithmische Modelltheorie I (Wahlpflichtfach)
Kennung	1113583
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Entscheidbare und unentscheidbare Theorien, Logik und Automaten, monadische Theorien, Prädikatenlogik auf endlichen Strukturen, Lokalität und Ehrenfeucht-Fraisse-Spiele, Fixpunktlogiken, TC Logiken, Logische Charakterisierung von Komplexitätsklassen, Interpretationen, automatische Strukturen, endlich präsentierbare Strukturen
Lernziele/Lernergebnisse	Verständnis der Zusammenhänge von logischer Definierbarkeit und algorithmischer Komplexität (Entscheidbarkeit von Theorien, Auswertungsalgorithmen, logische Charakterisierungen von Komplexitätsklassen). Beherrschen der modelltheoretischen und algorithmischen Methoden zur Analyse der Ausdrucksstärke und Komplexität logischer Spezifikationen auf endlichen und endlich präsentierbaren Strukturen. Fähigkeit, mit den fundamentalen Logiken der algorithmischen Modelltheorie umzugehen und diese in konkreten Szenarien anzuwenden.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Teilnahmebedingungen: Kenntnisse des Moduls Mathematische Logik I
Literatur	Skript zur Vorlesung; H.-D. Ebbinghaus, J. Flum, Finite Model Theory, Springer 1995; E. Grädel et al., Finite Model Theory and its Applications, Springer 2006; N. Immerman, Descriptive Complexity, Springer 1999; L. Libkin, Elements of Finite Model Theory, Springer 2004
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Zulassungsvoraussetzung: Lösen von Übungsaufgaben  Prüfungsleistung: Bestehen einer Klausur oder einer mündlichen Prüfung; Prüfungsdauer und -art werden am Anfang des Semesters bekannt gegeben
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. phil. Erich Grädel
ECTS Credits	9
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Modelltheorie I (1113583)

Gesamtstunden (h)	270,0
Präsenzstunden (h)	90,0
Selbststudium (h)	180,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Algorithmische Modelltheorie I (111358302)	1. Semester	keine Semesterempfehlung	0	2
Prüfungsleistung: Algorithmische Modelltheorie I (111358301)	1. Semester	keine Semesterempfehlung	9	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Algorithmische Modelltheorie I	1. Semester	keine Semesterempfehlung	-	4

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Spieltheorie (1212326)

Modultitel	Algorithmische Spieltheorie (Wahlpflichtfach)
Kennung	1212326
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	1. Einführung in die Spieltheorie; 2. Komplexität von spieltheoretischen Lösungskonzepten; 3. Kongestions- und Potentialspiel; 4. Der Preis der Anarchie; 5. Algorithmische Aspekte im Mechanismen Design
Lernziele/Lernergebnisse	Kenntnisse: Am Ende des Kurses sollen die Teilnehmer im Stande sein, fundamentale spieltheoretische Begriffe und Lösungskonzepte zu beschreiben und zu definieren, die Berechnungskomplexität von algorithmischen Problemen in der Spieltheorie und im Mechanismen Design zu analysieren und die wichtigsten Algorithmen für diese Probleme zu erklären.   Fertigkeiten: Die Teilnehmer sollen im Stande sein, spieltheoretische Annahmen kritisch zu diskutieren, Gleichgewichtspunkte und andere Lösungskonzepte zu analysieren, und Algorithmen zur Berechnung von Lösungskonzepten zu entwerfen und zu analysieren.   Kompetenzen: Die Teilnehmer sollen im Stande sein, strategische Situationen mit Hilfe von spieltheoretischen Begriffen und Konzepten zu modellieren und algorithmische Methoden zur Lösung von spieltheoretischen Problemen anzuwenden.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse in Algorithmen, diskreten Strukturen und Wahrscheinlichkeitstheorie (Stochastik).
Literatur	N. Nisan, T. Roughgarden, E. Tardos, V. Vazirani. Algorithmic Game Theory, Cambridge University Press, 2007.   T. Roughgarden. Selfish Routing and the Price of Anarchy. MIT Press, 2005.   A. Mas-Colell, M.D. Whinston, and J.R. Green. Microeconomic Theory. Oxford University Press, 1995.   M.J. Osborne. An Introduction to Game Theory. Oxford University Press. 2004.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Gerhard Wöginger
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Algorithmische Spieltheorie (1212326)

Selbststudium (h)

105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Algorithmic Game Theory (121232602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Algorithmic Game Theory (121232601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Algorithmic Game Theory	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Analyse von Algorithmen (1212337)

Modultitel	Analyse von Algorithmen (Wahlpflichtfach)
Kennung	1212337
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Die Vorlesung vermittelt die notwendigen Techniken, welche zur Analyse von Algorithmen benötigt werden, und wendet sie an zahlreichen Beispielen praktisch an. Dabei wird sowohl Wert darauf gelegt, die Algorithmen sehr genau zu analysieren - im Extremfall die genaue Anzahl von Maschineninstruktionen, die im Durchschnitt durchlaufen werden, zu bestimmen -, als auch grobe Abschätzungen mit minimalem Aufwand durchführen zu können.
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>Beim erfolgreichen Abschluss des Moduls sollten die Studierenden Kenntnisse über folgende Themen haben:</p> <ul style="list-style-type: none"> <li>• Mathematische Grundlagen für die Analyse von Algorithmen,</li> <li>• grundlegende Verfahren für das Lösen von Rekursionsgleichungen, insbesondere erzeugende Funktionen, Singularitätenanalyse und Sattelpunktmethode.</li> </ul> <p>Skills: Die Studierenden sollten in der Lage sein,</p> <ul style="list-style-type: none"> <li>• Algorithmen in ihre zu analysierenden Bestandteile zu zerlegen und entsprechende Rekursionsgleichungen aufzustellen,</li> <li>• einfache Algorithmen mithilfe der Standardmethoden präzise zu analysieren und das Ergebnis qualitativ zu interpretieren,</li> </ul> <p>Competences: Basierend auf dem Wissen und den Fähigkeiten sollten die Studierenden in der Lage sein</p> <ul style="list-style-type: none"> <li>• Effizienzprobleme beim Entwurf von Algorithmen vorherzusehen und entsprechend darauf zu reagieren,</li> <li>• bessere Programme zu schreiben, die in ihren laufzeitkritischen Bereichen hochoptimiert sind.</li> </ul>
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Wissen über Wahrscheinlichkeitsrechnung und elementare Algebra. Wissen im Bereich Effiziente Algorithmen.
Literatur	<p>Lecture Notes on Analysis of Algorithms and the books</p> <ul style="list-style-type: none"> <li>• R. Sedgewick and P. Flajolet. An Introduction to the Analysis of Algorithms.</li> <li>• R. Sedgewick and P. Flajolet. Analytic Combinatorics.</li> </ul>
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:</p>



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Analyse von Algorithmen (1212337)

	Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Universitätsprofessor Dr. rer. nat. Peter Rossmanith
ECTS Credits	8
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	0
Gesamtstunden (h)	240,0
Präsenzstunden (h)	90,0
Selbststudium (h)	150,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Analyse von Algorithmen (121233702)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Analyse von Algorithmen (121233701)	1. Semester	keine Semesterempfehlung	8	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Analyse von Algorithmen	1. Semester	keine Semesterempfehlung	-	4

Modultitel	Compilerbau (Wahlpflichtfach)
Kennung	1211978
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• Lexikalische Analyse von Programmen (Scanner)</li> <li>• Syntaktische Analyse von Programmen (Parser)</li> <li>• Semantische Analyse</li> <li>• Werkzeuge zur Compilerkonstruktion (lex, yacc)</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse</p> <ul style="list-style-type: none"> <li>• Kenntnisse über Methoden der Syntaxbeschreibung (reguläre Ausdrücke, kontextfreie und attributierte Grammatiken, EBNF)</li> <li>• Kenntnisse im Einsatz compilererzeugender Werkzeuge</li> </ul> <p>Fähigkeiten</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur Implementierung einfacher Compilerkomponenten (Scanner, Parser)</li> </ul> <p>Kompetenzen</p> <ul style="list-style-type: none"> <li>• Verständnis der Konstruktion und Wirkungsweise von Compilern für höhere Programmiersprachen</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Beherrschung der wesentlichen Konzepte imperativer und objektorientierter Programmiersprachen sowie elementarer Programmier Techniken in diesen Sprachen (Modul Programmierung). Kenntnis von Datenstrukturen wie Listen, Stacks, Queues und Bäumen (Modul Datenstrukturen und Algorithmen). Kenntnis grundlegender Automatenmodelle wie endliche Automaten und Kellerautomaten (Modul Formale Systeme, Automaten und Prozesse).
Literatur	<p>Folien und Skripte zur Vorlesung sowie folgende Lehrbücher:</p> <ul style="list-style-type: none"> <li>• A. Aho, R. Sethi, J. Ullman: Compilers – Principles, Techniques, and Tools. Addison-Wesley, 1988.</li> <li>• A.W. Appel, J. Palsberg: Modern Compiler Implementation in Java. Cambridge University Press, 2002.</li> <li>• D. Grune, H.E. Bal, C.J.H. Jacobs, K.G. Langendoen: Modern Compiler Design. Wiley &amp; Sons, 2000.</li> <li>• R. Wilhelm, D. Maurer: Übersetzerbau, 2. Auflage. Springer, 1997.</li> </ul>
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:          Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:          Dr. rer. nat. Katja PetzoldtModulverantwortlicher:</p>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Compilerbau (1211978)

	apl. Professor Dr. rer. nat. Thomas Noll Universitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Einführung in den Compilerbau (121197802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Einführung in den Compilerbau (121197801)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Einführung in den Compilerbau	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Concurrency Theory (1212646)

Modultitel	Concurrency Theory (Wahlpflichtfach)
Kennung	1212646
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Folgende Hauptthemen:</p> <ol style="list-style-type: none"> <li>1. Einführung</li> <li>2. Der "Interleaving"-Ansatz</li> <li>3. Der "True Concurrency"-Ansatz</li> <li>4. Verfeinerung und Kompositionalität</li> <li>5. Fallstudien und Softwarewerkzeuge</li> <li>6. Erweiterungen</li> </ol>
Lernziele/Lernergebnisse	<p>Erwerb der folgenden Kenntnisse, Fähigkeiten und Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Verständnis der theoretischen Grundlagen nebenläufiger Systeme</li> <li>• Fähigkeit zur formalen Modellierung und dem Vergleich solcher Systeme</li> <li>• Verständnis der wesentlichen semantischen Konzepte</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnis grundlegender Automatenmodelle (Vorlesung Formale Systeme, Automaten und Prozesse). Verständnis der Arbeitsweise paralleler und verteilter Systeme (Vorlesungen Betriebssysteme und Systemsoftware sowie Systemprogrammierung).
Literatur	<p>Folien und Skripte zur Vorlesung sowie folgende Lehrbücher:</p> <ul style="list-style-type: none"> <li>• Luca Aceto, Anna Ingólfssdóttir, Kim Guldstrand Larsen and Jiri Srba: Reactive Systems: Modelling, Specification and Verification. Cambridge University Press, 2007.</li> <li>• Maurice Herlihy and Nir Shavit: The Art of Multiprocessor Programming. Elsevier, 2008.</li> <li>• Jan Bergstra, Alban Ponse and Scott Smolka (Eds.): Handbook of Process Algebra. Elsevier, 2001.</li> <li>• Wolfgang Reisig: Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies. Springer Verlag, 2012.</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:  Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:  Dr. rer. nat. Katja Petzoldt  Modulverantwortlicher:  apl. Professor Dr. rer. nat. Thomas Noll  Universitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen</p>
ECTS Credits	6

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Concurrency Theory (1212646)

Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Concurrency Theory (121264602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Concurrency Theory (121264601)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Concurrency Theory	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Fixpoints and Induction in Logic and Computer Science (1226911)

Modultitel	Fixpoints and Induction in Logic and Computer Science (Wahlpflichtfach)
Kennung	1226911
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2022
Gültig bis	-
Modulniveau	Master
Inhalt	<p>This course is about the proof method of induction, and about the theory of fixpoints on ordered structures, complete lattices in particular. Topics covered are:</p> <ul style="list-style-type: none"> <li>Induction on natural numbers, structural induction, well-founded induction</li> <li>Tarski's fixpoint theorem for monotonic functions on complete lattices and variations on complete partial orders</li> <li>Examples from games, probability theory (Markov chains and decision processes), program semantics and verification, automata theory and formal languages</li> <li>Computation of fixpoints by iteration on finite lattices</li> <li>Characterization of fixpoints on infinite lattices for continuous functions and arbitrary functions</li> <li>Ordinal numbers, transfinite induction, well-ordering theorem</li> <li>First-order logic with fixpoints</li> <li>The role of induction in incompleteness theorem for Peano arithmetic</li> </ul>
Lernziele/Lernergebnisse	Students know the relation of inductive proofs and least fixpoints. They know the central role of fixpoints in various areas of computer science. They can identify order structures in different areas of computer science and discrete mathematics, in particular complete lattices, and can use the corresponding fixpoint theorems. They know and can apply strategies for computing fixpoints on finite lattices. They know the role of induction in incompleteness results for first-order logic.
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Keine.
Literatur	Will be announced in the lecture
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Dr. rer. nat. Christof Löding
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Fixpoints and Induction in Logic and Computer Science (1226911)

Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Klausur Fixpoints and Induction in Logic and Computer Science (122691101)	1. Semester	keine Semesterempfehlung	6	0
Übung Fixpoints and Induction in Logic and Computer Science (122691102)	1. Semester	keine Semesterempfehlung	0	3

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Fixpoints and Induction in Logic and Computer Science	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Graphalgorithmen (1212327)

Modultitel	Graphalgorithmen (Wahlpflichtfach)
Kennung	1212327
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Schnittgraphen, planare Graphen und andere Graphklassen, Baumweite, Bandweite und weitere Eigenschaften, Orientierungen auf Graphen, Perfekte Graphen, Einbettungen, Kommunikationsprobleme auf Graphen.
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>Beim erfolgreichen Abschluss des Moduls sollten die Studierenden Kenntnisse über folgende Themen haben Komplexität verschiedener Probleme auf verschiedenen Graphklassen.Algorithmen auf verschiedenen Graphklassen.</p> <p>Skills:</p> <p>Die Studierenden sollten in der Lage sein Graphklassen zu beschreiben,Beweise zu Grapheigenschaften zu führen.</p> <p>Competences:</p> <p>Basierend auf dem Wissen und den Fähigkeiten sollten die Studierenden in der Lage sein: Eigenschaften einer Graphklasse erkennen. Komplexität eines gegebenen Problems auf einer Graphklasse bestimmen.Algorithmen unter Ausnutzung der Eigenschaften einer Graphklasse entwickeln.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus den Modulen "Algorithmen und Datenstrukturen" sowie "Berechenbarkeit und Komplexität".
Literatur	Zur Vorlesung wird ein Skript erstellt und folgende Literatur empfohlen: Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs Harary F.: Graphentheorie, 1974. Wilson R.J.: Einführung in die Graphentheorie, 1972
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Dr. rer. nat. Walter Unger
ECTS Credits	6
Kontaktzeit (SWS)	5



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Graphalgorithmen (1212327)

Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Graphalgorithmen (121232702)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Graphalgorithmen (121232701)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Graphalgorithmen	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Graphzerlegungen und algorithmische Anwendungen (1212710)

Modultitel	Graphzerlegungen und algorithmische Anwendungen (Wahlpflichtfach)
Kennung	1212710
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>- Baumzerlegungen und Baumweite</li> <li>- Obstruktionen für Baumweite und Dualität</li> <li>- Algorithmen zum Berechnen von Baumzerlegungen</li> <li>- Dynamische Programmierung auf Baumzerlegungen</li> <li>- Große Baumweite und Expansion</li> <li>- Zusammenhangssysteme, Astzerlegungen, und Tangles</li> </ul>
Lernziele/Lernergebnisse	<p><b>Kenntnisse:</b> Kenntnis und Verständnis der mathematischen und algorithmischen Theorie von Graphzerlegungen und Weitenmaßen.</p> <p><b>Fähigkeiten:</b> Techniken zur Analyse der Struktur von Graphen und ähnlichen mathematischen Modellen.</p> <p>Techniken zur Entwicklung von Algorithmen, die auf strukturellen Zerlegungen beruhen.</p> <p><b>Kompetenzen:</b> Entwicklung eines tiefen Verständnisses für den Zusammenhang zwischen mathematischer Struktur und effizienten Algorithmen.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Gute Kenntnisse in den Bereichen Diskrete Mathematik, Berechenbarkeit und Komplexität, sowie Datenstrukturen und Algorithmen.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Martin Grohe</p>
ECTS Credits	6
Kontaktzeit (SWS)	5

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Graphzerlegungen und algorithmische Anwendungen (1212710)

Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Graphzerlegungen und algorithmische Anwendungen (121271002)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Graphzerlegungen und algorithmische Anwendungen (121271001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Graphzerlegungen und algorithmische Anwendungen	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Höhere Algorithmik (1223638)

Modultitel	Höhere Algorithmik (Wahlpflichtfach)
Kennung	1223638
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2021
Gültig bis	-
Modulniveau	Master
Inhalt	Die Vorlesung gibt eine Einführung in einige zentrale Teilgebiete der Höheren Algorithmik, wie zum Beispiel: Approximationsalgorithmen; Online Algorithmen; Meta-Heuristiken; Exakte Algorithmen; Parametrisierte Algorithmen; Effizient lösbare Spezialfälle; Randomisierte Algorithmen; Verteilte und Parallele Algorithmen; IO-effiziente Algorithmen. Jedes Teilgebiet bildet ein abgeschlossenes Kapitel, das die Hauptideen erklärt, grundlegende Definitionen und einige Musterresultate zeigt, und die Ziele und die Methodik zusammenfasst.
Lernziele/Lernergebnisse	<p><b>Kenntnisse:</b> Am Ende des Kurses sollen die Teilnehmer im Stande sein, die fundamentalen Begriffe und Konzepte der Höheren Algorithmik zu beschreiben und zu definieren, und die wichtigsten Algorithmen zu erklären.</p> <p><b>Fertigkeiten:</b> Die Teilnehmer sollen im Stande sein, algorithmische Probleme kritisch zu diskutieren und sie sollen die wichtigsten algorithmischen Methoden beherrschen.</p> <p><b>Kompetenzen:</b> Die Teilnehmer sollen im Stande sein, algorithmische Probleme zu modellieren und Methoden aus der Höheren Algorithmik zur Lösung von derartigen Problemen anzuwenden.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse in Datenstrukturen und Algorithmen; Effiziente Algorithmen; Berechenbarkeit und Komplexität;
Literatur	<p>T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein: Introduction to Algorithms, MIT Press, 2001.</p> <p>R. Motwani, P. Raghavan: Randomized Algorithms, Cambridge University Press, 1993.</p> <p>V. Vazirani: Approximation Algorithms, Springer, 2010.</p> <p>J. Flum, M. Grohe: Parameterized Complexity Theory, Springer, 2006.</p> <p>A. Borodin, R. El-Yaniv: Online Computation and Competitive Analysis, Cambridge University Press, 1998.</p> <p>F. Fomin, D. Kratsch: Exact Exponential Algorithms, Springer, 2010.</p> <p>E. Aarts, J.K. Lenstra: Local Search in Combinatorial Optimization, Princeton University Press, 2003.</p>
Sprache	Englisch
Prüfungsbedingungen	Klausur (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Höhere Algorithmik (1223638)

Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Klausur Höhere Algorithmik (122363801)	2. Semester	keine Semesterempfehlung	6	0
Übung Höhere Algorithmik (122363802)	2. Semester	keine Semesterempfehlung	0	2

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Höhere Algorithmik	2. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Infinite Computations and Games (1212336)

Modultitel	Infinite Computations and Games (Wahlpflichtfach)
Kennung	1212336
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2016
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>automata on infinite words (basic definitions and constructions)</li> <li>deterministic automata and classes of acceptance conditions</li> <li>infinite games as model of reactive systems</li> <li>automata on infinite trees and their connection to games</li> <li>applications in logic, verification, and synthesis</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse: Die Studierenden lernen Modelle für nichtterminierende Systeme kennen.</p> <p>Fähigkeiten: Die Studierenden erwerben die Fähigkeit, grundlegende Methoden zur Analyse und Konstruktion nichtterminierender Systeme anzuwenden.</p> <p>Kompetenzen: Die Studierenden kennen die Methoden und Modelle für zustandsbasierte nichtterminierende Systeme und können diese Anwenden.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Dr. rer. nat. Christof Löding
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Infinite Computations and Games (1212336)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Infinite Computations and Games (121233602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Infinite Computations and Games (121233601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Infinite Computations and Games	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Introduction to Algorithmic Differentiation (1221327)

Modultitel	Introduction to Algorithmic Differentiation (Wahlpflichtfach)
Kennung	1221327
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2021
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>We discuss the algorithmic differentiation of differentiable numerical programs, that is the generation of first-, second- and higher-order tangent and adjoint code by</p> <ul style="list-style-type: none"> <li>• operator and function overloading in C++</li> <li>• manual source transformation</li> <li>• automatic source transformation</li> </ul>
Lernziele/Lernergebnisse	<ul style="list-style-type: none"> <li>• Knowledge: understanding of fundamental algorithmic differentiation (AD) modes</li> <li>• Skills: ability to apply AD software to differentiable numerical programs</li> <li>• Competences: choice of appropriate AD mode for a given task</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>• Introduction to Programming</li> <li>• Introduction to Calculus</li> </ul>
Literatur	<ul style="list-style-type: none"> <li>• Set of slides</li> <li>• Example programs</li> <li>• Naumann: The Art of Differentiating Computer Programs. SIAM 2012.</li> <li>• Griewank, Walther: Evaluating Derivatives. SIAM 2008.</li> <li>• References to relevant current literature and online materials</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Uwe Naumann
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	120
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Introduction to Algorithmic Differentiation (1221327)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Introduction to Algorithmic Differentiation (122132702)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Introduction to Algorithmic Differentiation (122132701)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Einführung in Algorithmisches Differenzieren	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Introduction to Quantum Computing (1230246)

Modultitel	Introduction to Quantum Computing (Wahlpflichtfach)
Kennung	1230246
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2024
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>What are quantum computers? Are they magic computers that can compute everything faster? Or are they some science-fiction that has nothing to do with reality? How do they work? What can they do? In this lecture, we will introduce the basics of quantum computing from the computer science perspective. We will understand how quantum mechanics allows us to compute some problems faster (and some not), how to program a quantum computer, and what the challenges are. There will be a mix of mathematical foundations, and applied quantum computer programming.</p> <p>Content:</p> <ul style="list-style-type: none"> <li>• Mathematical foundations of quantum mechanics</li> <li>• How do quantum computers work?</li> <li>• How are quantum computers built? (Physical implementations.)</li> <li>• Why are quantum computers difficult (problems with noise)</li> <li>• Programming a quantum computer (Writing your own quantum programs.)</li> <li>• Quantum algorithms: <ul style="list-style-type: none"> <li>• Simple basic algorithms</li> <li>• Grover's algorithm ("database" search)</li> <li>• Shor's algorithm (factoring, etc.)</li> <li>• Possibly more</li> </ul> </li> <li>• Quantum simulation (simulating physical processes with a quantum computer)</li> <li>• The course comes with both theory and with programming exercises, to run on simulated (or real) quantum computers.</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: After successful completion of the module students know:</p> <ul style="list-style-type: none"> <li>• the basic mathematics of quantum computing</li> <li>• how quantum computers work</li> <li>• how quantum computers are built</li> <li>• how the most important quantum algorithms work</li> <li>• what the problems are with building quantum computers</li> <li>• how quantum computers are programmed</li> <li>• how quantum computers relate to the simulation of physical systems</li> </ul> <p>Skills: After successful completion of the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• compute what a quantum program does (on paper)</li> <li>• program quantum computers (simple programs)</li> <li>• understand research papers introducing quantum algorithms</li> <li>• understand why quantum computers cannot just do everything faster</li> </ul> <p>Competencies: Based on the knowledge and skills acquired in the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• understand claims about quantum computers (recognize obvious nonsense)</li> <li>• put modern developments in quantum computing into context</li> <li>• follow advanced quantum computing related courses</li> <li>• join projects working on the software-side of quantum computing</li> </ul>
Teilnahmebedingungen (studiengangspezifisch)	Keine

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Introduction to Quantum Computing (1230246)

(empfohlene) Voraussetzungen	Keine.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Univ.-Prof. Dr. rer. nat. Dominique Unruh
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Introduction to Quantum Computing (123024601)	2. Semester	keine Semesterempfehlung	6	0
Übung Introduction to Quantum Computing (123024602)	2. Semester	keine Semesterempfehlung	0	2

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Introduction to Quantum Computing	2. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Komplexitätstheorie (1212331)

Modultitel	Komplexitätstheorie (Wahlpflichtfach)
Kennung	1212331
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Deterministische, nichtdeterministische, probabilistische und parallele Berechnungsmodelle; Komplexitätsklassen; Reduktionen; Komplexität von Approximationsproblemen; Auswahl fortgeschrittener Themen wie Interaktive Beweise, Derandomisierung, Schaltkreiskomplexität, Kommunikationskomplexität, parametrische Komplexität.
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>- Vertiefende Kenntnisse in den Kerngebieten der Komplexitätstheorie</li> </ul> <p>Fähigkeiten:</p> <ul style="list-style-type: none"> <li>- Die Fähigkeit, algorithmische Probleme bezüglich Ihrer Komplexität zu klassifizieren</li> <li>- Die Fähigkeit, das Verhältnis zwischen verschiedenen Komplexitätsklassen zu analysieren</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>- Tiefergehendes Verständnis für die wichtigsten Komplexitätsklassen und Komplexitätsmaße und das Wechselspiel zwischen verschiedenen Aspekten von Komplexität</li> </ul>
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Mathematische Grundlagen in den Bereichen Diskrete Strukturen und Lineare Algebra. Informatische Grundlagen in den Bereichen Datenstrukturen und Algorithmen, Berechenbarkeit und Komplexität, und Logik.
Literatur	Arora, Barak: Computational Complexity – A Modern Approach; Papadimitriou: Computational Complexity; Wegener: Komplexitätstheorie
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:</p> <p>Dr. rer. nat. Katja PetzoldtModulverantwortlicher:</p>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Komplexitätstheorie (1212331)

	Universitätsprofessor Dr. rer. nat. Martin Grohe
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	90
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Komplexitätstheorie (121233102)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Komplexitätstheorie (121233101)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Komplexitätstheorie	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Komplexitätstheorie und Quantum Computing (1113594)

Modultitel	Komplexitätstheorie und Quantum Computing (Wahlpflichtfach)
Kennung	1113594
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Zweimestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2006
Gültig bis	-
Modulniveau	Master
Inhalt	Deterministische, nichtdeterministische, parallele und probabilistische Berechnungsmodelle und die zugehörigen Komplexitätsklassen, vollständige Probleme, Komplexitätstheorie für Optimierungsprobleme, Logik und Komplexität, Einführung in die mathematischen und physikalischen Grundlagen des Quantum Computing, Quantenbits und Quantenregister, Quantum Gate Arrays, wichtige Quantenalgorithmen, insbesondere der Faktorisierungsalgorithmus von Shor, Quanteninformationstheorie
Lernziele/Lernergebnisse	Die Studierenden sollen in der Lage sein, algorithmische Probleme bezüglich ihrer Komplexität zu klassifizieren. Sie sollen die wichtigsten Komplexitätsklassen für deterministische, nichtdeterministische, parallele und probabilistische Berechnungsmodelle kennen und ihre Zusammenhänge verstehen. Die Studierenden sollen die Grundlagen und wichtigsten Algorithmen des Quantum Computing beherrschen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse der Module Algebra, Berechenbarkeit und Komplexität
Literatur	Skript zur Vorlesung;  C. Papadimitriou, Computational Complexity, Addison Wesley 1994;  M. Hirvensalo, Quantum Computing, Springer, 2001;  M. Nielsen and I. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Zulassungsvoraussetzung: Lösen von Übungsaufgaben  Prüfungsleistung: Bestehen einer Klausur oder einer mündlichen Prüfung; Prüfungsdauer und -art werden am Anfang des Semesters bekannt gegeben
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. phil. Erich Grädel
ECTS Credits	9
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Komplexitätstheorie und Quantum Computing (1113594)

Gesamtstunden (h)	270,0
Präsenzstunden (h)	90,0
Selbststudium (h)	180,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Komplexitätstheorie und Quantum Computing (111359402)	1. Semester	keine Semesterempfehlung	0	1
Prüfungsleistung: Komplexitätstheorie und Quantum Computing (111359401)	1. Semester	keine Semesterempfehlung	9	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Komplexitätstheorie und Quantum Computing	1. Semester	keine Semesterempfehlung	-	2

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Logik und Spiele (1113599)

Modultitel	Logik und Spiele (Wahlpflichtfach)
Kennung	1113599
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Fundamentale Modelle und Begriffe der Spieltheorie, Endliche und unendliche Spiele, Model-Checking-Spiele, Determinierte und nichtdeterminierte Spiele, Borel-Spiele, Muller-Spiele und Paritätsspiele, Komplexität und Definierbarkeit von Gewinnregionen, Algorithmische Synthese und Optimierung von Gewinnstrategien, Mehrpersonenspiele
Lernziele/Lernergebnisse	Verständnis der grundlegenden Begriffe und Probleme der algorithmischen Spieltheorie und der Zusammenhänge von Logik und Spieltheorie, Kenntnis der logischen und algorithmischen Methoden zur Behandlung unendlicher Spiele, Verständnis der Anwendungen unendlicher Spiele als Modell reaktiver Systeme und zur Auswertung logischer Formeln
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus dem Modul Mathematische Logik oder vergleichbare Leistungen
Literatur	Skript zur Vorlesung; E. Grädel, W. Thomas, Th. Wilke (Eds.), Automata, Logics and Infinite Games, Springer 2002; M. Osborne, A. Rubinstein, A Course in Game Theory, The MIT Press, Cambridge, MA 1994
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Zulassungsvoraussetzung: Lösen von Übungsaufgaben  Prüfungsleistung: Bestehen einer Klausur oder einer mündlichen Prüfung; Prüfungsdauer und -art werden am Anfang des Semesters bekannt gegeben
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. phil. Erich Grädel
ECTS Credits	9
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-
Gesamtstunden (h)	270,0
Präsenzstunden (h)	90,0
Selbststudium (h)	180,0



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Logik und Spiele (1113599)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Logik und Spiele (111359902)	1. Semester	keine Semesterempfehlung	0	2
Prüfungsleistung: Logik und Spiele (111359901)	1. Semester	keine Semesterempfehlung	9	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Logik und Spiele	1. Semester	keine Semesterempfehlung	-	4

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Mathematische Logik II (1112957)

Modultitel	Mathematische Logik II (Wahlpflichtfach)
Kennung	1112957
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Mengenlehre und Grundlagen der Mathematik, Ordinalzahlen und Kardinalzahlen, Auswahlaxiom, Gödelsche Unvollständigkeitssätze, Einführung in die Modelltheorie, Fixpunktlogiken
Lernziele/Lernergebnisse	Die Studierenden sollen Verständnis für die Grundlagenprobleme der Mathematik (und Informatik) entwickeln und die Möglichkeiten und Grenzen der mengentheoretischen Fundierung der Mathematik auf der Grundlage des Axiomensystems ZFC verstehen. Die im Modul Mathematische Logik eingeführten Methoden und Werkzeuge sollen vertieft und erweitert werden. Insbesondere sollen die Studierenden in die Lage versetzt werden, mit Ordinalzahlen und transfiniter Induktion sowie mit grundlegenden modelltheoretischen Methoden umzugehen. Über die im Modul Mathematische Logik behandelten logischen Systeme hinaus wird ein besonderes Gewicht auf Fixpunktlogiken (Mu-Kalkül und LFP) gelegt. Ziel ist ein Verständnis der Ausdrucksstärke solcher Formalismen und die Fähigkeit, mathematische Sachverhalte in Fixpunktlogiken auszudrücken.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Teilnahmebedingungen: Kenntnisse des Moduls Mathematische Logik I
Literatur	-
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Zulassungsvoraussetzung: Lösen von Übungsaufgaben  Prüfungsleistung: Bestehen einer Klausur oder einer mündlichen Prüfung; Prüfungsdauer und -art werden am Anfang des Semesters bekannt gegeben
Sonstiges	-
Modulverantwortung	N.N.
ECTS Credits	9
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-
Gesamtstunden (h)	270,0
Präsenzstunden (h)	90,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Mathematische Logik II (1112957)

Selbststudium (h)

180,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Mathematische Logik II (111295702)	1. Semester	keine Semesterempfehlung	0	2
Prüfungsleistung: Mathematische Logik II (111295701)	1. Semester	keine Semesterempfehlung	9	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Mathematische Logik II	1. Semester	keine Semesterempfehlung	-	4

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Modeling and Verification of Probabilistic Systems (1212711)

Modultitel	Modeling and Verification of Probabilistic Systems (Wahlpflichtfach)
Kennung	1212711
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>Markov chains and Markov decision processes.</li> <li>Parallel composition and restriction operators.</li> <li>(Strong and weak) bisimulation and simulation relations.</li> <li>Reachability and omega-regular objectives.</li> <li>Probabilistic extensions of temporal logics.</li> <li>Counterexample generation algorithms.</li> <li>Interactive Markov chains and probabilistic automata.</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: on completion of this course, students have acquired basic knowledge about</p> <ul style="list-style-type: none"> <li>Concurrent systems</li> <li>Process algebras</li> <li>Markov chains and probabilistic automata</li> <li>The interplay between probability and concurrency</li> <li>Bisimulation, congruence, and axiomatisation</li> </ul> <p>Skills: on completion of this course, students are skilled to</p> <ul style="list-style-type: none"> <li>Model randomised distributed systems in process algebra</li> <li>Manipulate process algebraic descriptions</li> </ul> <p>Competences: on completion of this course, students are able to</p> <ul style="list-style-type: none"> <li>Read, understand, and write process algebraic specifications</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basic knowledge about algorithms, logics, automata theory, and probability theory.
Literatur	<p>Lecture slides</p> <p>Recommended books:</p> <ul style="list-style-type: none"> <li>Baier, Katoen: Principles of Model Checking, MIT Press, 2008.</li> <li>Hermanns, Interactive Markov Chains, Springer, 2002.</li> <li>Brinksma, Hermanns, Katoen (eds): Lectures on Formal Methods and Performance Analysis. LNCS 2090, 2001.</li> <li>Tijms: A First Course in Stochastic Models. Wiley, 2003.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:</p> <p>Dr. rer. nat. Katja Petzoldt</p> <p>Modulverantwortlicher:</p> <p>Universitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen</p>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Modeling and Verification of Probabilistic Systems (1212711)

ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Modeling and Verification of Probabilistic Systems (121271102)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Modeling and Verification of Probabilistic Systems (121271101)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Modeling and Verification of Probabilistic Systems	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Modellierung und Analyse hybrider Systeme (1212339)

Modultitel	Modellierung und Analyse hybrider Systeme (Wahlpflichtfach)
Kennung	1212339
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Hybrid systems are systems with mixed discrete and continuous behaviour. Typical examples are physical systems which continuously evolve over time and which are controlled by some discrete controller, e.g., a chip or a computer.</p> <p>The behaviour of hybrid systems is often safety-critical. For example, in case of an accident an airbag can save the life of the car driver, but only if the airbag reacts in time. To assure the correct functioning of such safety-critical hybrid systems, their automatic synthesis and analysis is of high importance.</p> <p>In the lecture we first introduce hybrid automata to model hybrid systems and logics to specify safety and liveness properties of the models. We introduce different classes of hybrid automata with increasing expressive power. For each class we discuss whether the reachability problem is decidable, and develop algorithms for their analysis. Finally we discuss methods for abstraction and for the over-approximative representation of state sets and show how they can be used for reachability analysis.</p> <p>Contents:</p> <ul style="list-style-type: none"> <li>• Discrete, continuous and hybrid systems, examples</li> <li>• Modeling formalism: hybrid automata</li> <li>• Some important features: time determinism, time divergence, Zeno-behaviour, stability etc.</li> <li>• Interesting classes of hybrid automata: timed automata, rectangular automata, linear hybrid automata, non-linear hybrid automata</li> <li>• Analysis: model checking, abstraction, approximation</li> </ul>
Lernziele/Lernergebnisse	<p><b>Knowledge:</b> In this lecture the students will learn modeling formalisms for discrete, continuous and hybrid systems, formalisms to specify relevant properties of those models, and gain knowledge in the areas of verification, abstraction and approximation algorithms to prove or disprove the validity of those properties. They will get known to different classes of hybrid automata and learn the advantages and disadvantages of their expressive power including decidability results.</p> <p><b>Skills:</b> The students collect experiences in the application of the above formalisms to build models of real-world systems at different abstraction levels and to formalize and prove their correct functioning. For complex or undecidable problems they will be able to apply safe abstraction and approximation techniques.</p> <p>Interactive learning methods help to increase the interest in theoretical computer science and to improve communication skills (e.g., the verbal formalization of scientific problems).</p> <p><b>Competences:</b> The students will know when and how they can apply formal techniques during the development of complex systems. They will train logical and analytical thinking, especially for the development of complex safety-critical systems involving physical components. They will recognize the importance of the application of formal methods in these procedures.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Modellierung und Analyse hybrider Systeme (1212339)

(empfohlene) Voraussetzungen	Keine.
Literatur	To be announced in the lecture
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Universitätsprofessorin Dr. Erika Abraham
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Modellierung und Analyse hybrider Systeme (121233902)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Modellierung und Analyse hybrider Systeme (121233901)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Modellierung und Analyse hybrider Systeme	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Near-term Quantum Computation (1230345)

Modultitel	Near-term Quantum Computation (Wahlpflichtfach)
Kennung	1230345
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	<p>This lecture is an interdisciplinary (computer science, physics, mathematics) introduction to modern algorithmic challenges as they occur in the usage of near-term quantum computing hardware. After reviewing the basics of quantum computation and algorithms we will cover the limitation of current quantum computing hardware from a computer scientist's point of view. Next, we will discuss algorithms for near-term quantum computers, before we turn to quantum compilation, i.e. algorithms for the optimal mapping of quantum algorithms to real quantum computers. Finally, we will cover the basics of quantum error correction and mitigation.</p> <p>This lecture is an interdisciplinary (physics, computer science, mathematics) introduction to quantum algorithms for near-term quantum computing hardware. After reviewing the basics of quantum computation and algorithms we will cover the limitation of current quantum computing hardware. Next, we will discuss algorithms for near-term quantum computers, which is the main focus of the lecture. Finally, we turn to quantum compilation, i.e. algorithms for the optimal mapping of quantum algorithms to real quantum computers, as well as quantum error mitigation and correction.</p>
Lernziele/Lernergebnisse	<p><b>Knowledge</b> After successful completion of the module students know:</p> <ul style="list-style-type: none"> <li>• basics and limitations variational quantum algorithms on near-term (noisy) quantum computers as well as quantum annealers</li> <li>• how to map applications to near-term (noisy) quantum computers</li> <li>• basic concepts of quantum compilation</li> </ul> <p><b>Skills</b> After successful completion of the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• implement quantum algorithms on real quantum computing devices</li> <li>• assess the computational capabilities of near-term (noisy) quantum computers</li> <li>• represent and manipulate quantum circuits with the diagrammatic calculus (ZX-calculus)</li> </ul> <p><b>Competencies</b> Based on the knowledge and skills acquired in the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• orient themselves in field of near-term (noisy) quantum computation</li> <li>• perform research on application focused quantum algorithms for near-term quantum computers</li> <li>• perform research on quantum compilation for early quantum computers</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Linear Algebra; Quantum Information/Computing Fundamentals
Literatur	-
Sprache	Englisch



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Near-term Quantum Computation (1230345)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Univ.-Prof. Dr. rer. nat. Dominique Unruh
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Near-term Quantum Computation (123034501)	2. Semester	keine Semesterempfehlung	6	0
Übung Near-term Quantum Computation (123034502)	2. Semester	keine Semesterempfehlung	0	1

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Near-term Quantum Computation	2. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Online Algorithmen (1212645)

Modultitel	Online Algorithmen (Wahlpflichtfach)
Kennung	1212645
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Eine Auswahl von wichtigen Online-Problemen, wie: Paging, k-Server, List-Access, Daten-Management, Rucksack, Job-Scheduling, Graph Problemen, Bit Guessing. Sowie Grundlagen der Randomisierung und Advice-Komplexität.
Lernziele/Lernergebnisse	Die Kenntnis über die theoretischen Grundlagen von Online-Algorithmen mit einem Fokus auf algorithmische Problemstellungen und Lösungen.   Die Fähigkeit zur Modellierung von algorithmischen Problemen bei Online-Problemen   Kenntnis über grundlegende algorithmische Entwicklungsprinzipien für Online-Probleme.   Die Kompetenz mit den typischen Techniken für Online-Problemen obere und untere Schranken zu Online-Problemen zu entwickeln.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse über Algorithmen, diskrete Strukturen und Wahrscheinlichkeitstheorie.
Literatur	Dennis Komm, An introduction to online computation: determinism, randomization, advice, Springer 2015   A. Borodin, R. El-Yaniv: Online Computation and Competitive Analysis, Cambridge University Press 2005
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Gerhard Wöginger & Dr. rer. nat. Walter Unger
ECTS Credits	6
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Online Algorithmen (1212645)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Online Algorithmen (121264503)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Online Algorithmen (121264502)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Online Algorithmen	1. Semester	keine Semesterempfehlung	-	3
Erweiterte Vorlesung Online Algorithmen	1. Semester	keine Semesterempfehlung	-	1

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Parametrisierte Algorithmen (1212338)

Modultitel	Parametrisierte Algorithmen (Wahlpflichtfach)
Kennung	1212338
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Viele Probleme, die in der Praxis auftauchen, sind NP-hart und werden daher von Theoretikern in der klassischen Sichtweise als praktisch nicht exakt lösbar eingestuft.</p> <p>Parametrisierte Algorithmen versuchen gezielt, die Einfachheit praktisch vorkommender Probleminstanzen auszunutzen. In dieser Vorlesung werden parametrisierte Algorithmen und allgemeine Techniken, solche Algorithmen zu entwerfen, vorgestellt. Der Schwerpunkt liegt dabei auf jenen Techniken, die praktisch verwertbare Algorithmen hervorbringen. Daneben gibt es auch Verfahren, um nur zu zeigen, daß ein Problem einen parametrisierten Algorithmus besitzt, ohne zunächst auf die Effizienz zu achten. Schließlich gibt es Methoden, die nachweisen, daß ein Problem wahrscheinlich auch parametrisiert nicht schnell gelöst werden kann.</p>
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>Beim erfolgreichen Abschluss des Moduls sollten die Studierenden Kenntnisse über folgende Themen haben:</p> <ul style="list-style-type: none"> <li>• Grundlagen der parametrisierten Komplexitätstheorie,</li> <li>• Parametrisierte Algorithmen und Entwurfsmethoden für wichtige Probleme.</li> </ul> <p>Skills:</p> <p>Die Studierenden sollten in der Lage sein,</p> <ul style="list-style-type: none"> <li>• Die Methoden bounded search trees, iterative compression, Baumweite-basierte Verfahren, color coding, kernels und andere Standardverfahren auf ein geeignetes Problem anzuwenden.</li> <li>• nachweisen zu können, daß ein Problem wahrscheinlich nicht durch einen parametrisierten Algorithmus gelöst werden kann</li> </ul> <p>Competences:</p> <p>Basierend auf dem Wissen und den Fähigkeiten sollten die Studierenden in der Lage sein</p> <ul style="list-style-type: none"> <li>• festzustellen, welches Verfahren für ein parametrisiertes Problem am besten geeignet ist und es auf diese Weise zu lösen,</li> <li>• neue parametrisierte Algorithmen zu entwickeln, falls keine Standardverfahren anwendbar ist.</li> <li>• Für schwere Probleme der Praxis geeignete Parameter zu identifizieren und das Problem mit ihrer Hilfe effizient zu lösen.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse in Effiziente Algorithmen
Literatur	<ul style="list-style-type: none"> <li>• R. Downey and M. Fellows. Parameterized Algorithms.</li> <li>• R. Niedermeier. Invitation to Fixed-Parameter Algorithms.</li> <li>• J. Flum and M. Grohe. Parameterized Complexity Theory.</li> </ul>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Parametrisierte Algorithmen (1212338)

Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik</p> <p>Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt</p> <p>Modulverantwortlicher: Universitätsprofessor Dr. rer. nat. Peter Rossmanith</p>
ECTS Credits	8
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	0
Gesamtstunden (h)	240,0
Präsenzstunden (h)	90,0
Selbststudium (h)	150,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Parametrisierte Algorithmen (121233802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Parametrisierte Algorithmen (121233801)	1. Semester	keine Semesterempfehlung	8	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Parametrisierte Algorithmen	1. Semester	keine Semesterempfehlung	-	4

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Post-quantum cryptography (1229347)

Modultitel	Post-quantum cryptography (Wahlpflichtfach)
Kennung	1229347
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Foundations <ul style="list-style-type: none"> <li>• Mathematical foundations of post-quantum security</li> <li>• Quantum computing (programming quantum computers)</li> <li>• Cryptographic security proofs</li> <li>• Quantum random-oracle model</li> </ul> </li> <li>• Insecurity of existing cryptography <ul style="list-style-type: none"> <li>• RSA / discrete logarithm based cryptosystems</li> <li>• Shor's algorithm</li> </ul> </li> <li>• Elementary post-quantum schemes <ul style="list-style-type: none"> <li>• Learning with errors (LWE) problem</li> <li>• Decoding problems</li> <li>• Regev cryptosystems and variants</li> <li>• McEliece and variants</li> </ul> </li> <li>• Hardening encryption schemes with hash functions <ul style="list-style-type: none"> <li>• Quantum random-oracle model (QROM)</li> <li>• Fujisaki-Okamoto transform</li> <li>• Security proofs in the QROM</li> </ul> </li> <li>• Real-world post-quantum secure systems <ul style="list-style-type: none"> <li>• Overview over NIST-standardization process</li> <li>• One cryptosystem in detail (e.g., Kyber)</li> </ul> </li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: After successful completion of the module students know:</p> <ul style="list-style-type: none"> <li>• how quantum computers threaten cryptography</li> <li>• the techniques underlying modern quantum-safe cryptography</li> <li>• how security proofs work</li> <li>• important proof techniques for post-quantum cryptography</li> <li>• the current state of post-quantum security standardization</li> <li>• at least one standardization-candidate cryptosystem in more detail</li> </ul> <p>Skills: After successful completion of the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• write simple post-quantum security proofs</li> <li>• follow complex security proofs</li> <li>• connect complex (real-life) post-quantum secure cryptosystems with their basic versions</li> <li>• understand security-relevant design decisions</li> <li>• implement post-quantum cryptosystems</li> </ul> <p>Competencies: ;Based on the knowledge and skills acquired in the module, students will be able to:</p> <ul style="list-style-type: none"> <li>• orient themselves in the post-quantum security landscape based on solid foundations</li> <li>• enter research into post-quantum security</li> <li>• follow both industry and academic research, development and standardization of post-quantum crypto</li> <li>• make educated decisions related to cryptographic systems</li> <li>• join teams working on cryptographic systems (research / implementation)</li> </ul>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Post-quantum cryptography (1229347)

Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Introduction to Quantum Computing or comparable.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. ;Dominique Unruh
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	90
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Post-quantum cryptography (122934701)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung/Übung Postquantum cryptography	1. Semester	keine Semesterempfehlung	-	5

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Probabilistic Programming (1212650)

Modultitel	Probabilistic Programming (Wahlpflichtfach)
Kennung	1212650
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Main topics:</p> <ul style="list-style-type: none"> <li>• Probabilistic programming languages</li> <li>• Example applications of probabilistic programs</li> <li>• Probabilistic inference</li> <li>• Formal semantics</li> <li>• Conditioning</li> <li>• Program transformations</li> <li>• Loop invariants</li> <li>• Notions of termination of probabilistic programs</li> <li>• Expected run-time analysis of probabilistic programs</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: on completion, students have acquired detailed knowledge on:</p> <ul style="list-style-type: none"> <li>• Hoare-style reasoning</li> <li>• Markov decision processes</li> <li>• Bayesian networks</li> <li>• Hardness of almost-sure termination</li> </ul> <p>Skills: on completion of this course, students are skilled to:</p> <ul style="list-style-type: none"> <li>• Write and execute elementary probabilistic programs</li> <li>• Understand the main concepts behind probabilistic programming</li> <li>• Reasoning with weakest preconditions, loop invariants etc.</li> </ul> <p>Competences: on completion of this course, students are able to:</p> <ul style="list-style-type: none"> <li>• Judge the applicability of probabilistic programming</li> <li>• Apply various methods to program analysis</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basic knowledge of probability theory, basic programming language paradigms, logic and automata theory
Literatur	<p>In addition to the slides, the books:</p> <ul style="list-style-type: none"> <li>• A. McIver and C.C. Morgan. Abstraction, Refinement and Proof for Probabilistic Systems, Springer, 2004.</li> <li>• A. Pfeffer. Practical Probabilistic Programming. Manning. 2016.</li> <li>• N. Goodman and A. Stuhlmüller. The Design and Implementation of Probabilistic Programming Languages. <a href="http://dippl.org">http://dippl.org</a>, 2016.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:  Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:  Dr. rer. nat. Katja PetzoldtModulverantwortlicher:</p>



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Probabilistic Programming (1212650)

	Universitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Probabilistic Programming (121265002)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Probabilistic Programming (121265001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Probabilistic Programming	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Quantum Computing (1114998)

Modultitel	Quantum Computing (Wahlpflichtfach)
Kennung	1114998
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Einführung in die mathematischen Grundlagen des Quantum Computing, Quantenbits und Quantenregister, Quantum Gate Arrays. Wichtige Quantenalgorithmen, insbesondere der Faktorisierungsalgorithmus von Shor. Quanteninformationstheorie
Lernziele/Lernergebnisse	<p>Wissen</p> <ul style="list-style-type: none"> <li>• Theoretische Grundlagen und wichtigste Algorithmen des Quantum Computing</li> </ul> <p>Fähigkeiten</p> <ul style="list-style-type: none"> <li>• Analyse von Quanten-Algorithmen.</li> </ul> <p>Kompetenzen</p> <ul style="list-style-type: none"> <li>• Verständnis der Grundlagen des Quantum Computing und Einschätzen der Möglichkeiten und Grenzen des Quantum Computing.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine
Literatur	<p>M. Hirvensalo, Quantum Computing, Springer, 2001.</p> <p>M. Nielsen and I. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000</p> <p>David Mermin, Quantum Computer Science. An Introduction. Cambridge University Press, 2007</p>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	<p>Zulassungsvoraussetzung: Lösen von Übungsaufgaben</p> <p>Die Benotung ergibt sich zu 100% aus der abschließenden Prüfung zum Modul. Wird vorgesehen, dass semesterbegleitende Hausaufgaben auf die Prüfungsnote angerechnet werden, sind die relevanten Regelungen der Prüfungsordnung zu beachten.</p>
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. phil. Erich Grädel
ECTS Credits	4
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Quantum Computing (1114998)

Gesamtstunden (h)	120,0
Präsenzstunden (h)	60,0
Selbststudium (h)	60,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Quantum Computing (111499802)	2. Semester	keine Semesterempfehlung	0	2
Masterprüfung Quantum Computing (111499801)	2. Semester	keine Semesterempfehlung	4	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Quantum Computing	2. Semester	keine Semesterempfehlung	-	2

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Rekursionstheorie (1212335)

Modultitel	Rekursionstheorie (Wahlpflichtfach)
Kennung	1212335
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Charakterisierungen der berechenbaren Funktionen und Prädikate; Nummerierung berechenbarer Funktionen und universelle Funktionen; Unentscheidbarkeit und rekursive Aufzählbarkeit; Kleene'scher Fixpunktsatz und Rekursionssätze; Reduktionsbegriffe; die Struktur rekursiv aufzählbarer Mengen; die arithmetische Hierarchie; berechenbare Funktionale
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Vertiefende Kenntnisse der klassischen Berechenbarkeitstheorie</li> </ul> <p>Fähigkeiten:</p> <ul style="list-style-type: none"> <li>• die Fähigkeit, verschiedene mathematische Modelle zu verwenden und ineinander zu übersetzen</li> <li>• die Fähigkeit, algorithmische Probleme in grundlegende Berechenbarkeitsklassen einzuordnen</li> <li>• die Fähigkeit, fortgeschrittene Rekursionstechniken anzuwenden</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Tiefgehendes Verständnis für Berechenbarkeit und die Grenzen der Berechenbarkeit</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Logik, Datenstrukturen und Algorithmen, Berechenbarkeit und Komplexität
Literatur	<ul style="list-style-type: none"> <li>• Cutland, N. (1980). Computability. Cambridge University Press.</li> <li>• Odifreddi, P. (1992a). Classical Recursion Theory, Volume I. Elsevier.</li> <li>• Rogers, H. (1967). Theory of Recursive Functions and Effective Computability. McGraw-Hill.</li> </ul>
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Martin Grohe
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Rekursionstheorie (1212335)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Rekursionstheorie (121233502)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Rekursionstheorie (121233501)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Rekursionstheorie	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Semantik und Verifikation von Software (1212329)

Modultitel	Semantik und Verifikation von Software (Wahlpflichtfach)
Kennung	1212329
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>Introduction of WHILE model language</li> <li>Operational, denotational, and axiomatic semantics of WHILE</li> <li>Equivalence of operational and denotational semantics</li> <li>Dataflow analysis and abstract interpretation</li> <li>Abstraction and refinement</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge</p> <p>on successful completion of this module, students should be able to</p> <ul style="list-style-type: none"> <li>illustrate the fundamental concepts of formal semantics for imperative programming languages</li> <li>explain the differences between operational, denotational, and axiomatic semantics</li> <li>describe connections between different kinds of formal semantics</li> </ul> <p>Skills</p> <p>on successful completion of this module, students should be able to</p> <ul style="list-style-type: none"> <li>reason about the behaviour and correctness of programs using formal derivation and proof systems</li> <li>apply formal concepts for proving the correctness of software, program analyses and optimisations, and compilers</li> <li>develop semantic models for advanced programming language features such as procedures or threads</li> </ul> <p>Competences</p> <p>on successful completion of this module, students should be able to apply formal reasoning techniques in the development of computer software</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Understanding essential concepts of imperative and object-oriented programming languages and elementary programming techniques. Knowledge of foundations of formal systems and automata theory. Fundamental knowledge of mathematical logic.
Literatur	<p>Folien und Skripte zur Vorlesung sowie folgende Lehrbücher:</p> <ul style="list-style-type: none"> <li>G. Winskel: The Formal Semantics of Programming Languages. MIT Press, 1993.</li> <li>H.R. Nielson, F. Nielson: Semantics with Applications: A Formal Introduction, Wiley, 1992.</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Semantik und Verifikation von Software (1212329)

Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik  Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt  Modulverantwortlicher: apl. Professor Dr. rer. nat. Thomas Noll
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Semantik und Verifikation von Software (121232902)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Semantik und Verifikation von Software (121232901)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Semantik und Verifikation von Software	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Statische Programmanalyse (1212330)

Modultitel	Statische Programmanalyse (Wahlpflichtfach)
Kennung	1212330
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>Datenflussanalyse</li> <li>Abstrakte Interpretation</li> <li>Constraint-basierte Analyse</li> <li>Interprozedurale Analyse</li> <li>Analyse von Pointerprogrammen</li> <li>Anwendungen (Programmverifikation, optimierende Compiler)</li> </ul>
Lernziele/Lernergebnisse	<p>Erwerb der folgenden Kenntnisse, Fähigkeiten und Kompetenzen:</p> <ul style="list-style-type: none"> <li>Kenntnisse der Grundlagen der statischen Analyse von Computerprogrammen</li> <li>Fähigkeit zur Konstruktion und zum Einsatz von Werkzeugen zur Programmanalyse</li> <li>Kompetenz zur Darstellung von Korrektheitseigenschaften von Software als Programmanalyseprobleme</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Beherrschung der wesentlichen Konzepte imperativer und objektorientierter Programmiersprachen sowie elementarer Programmier Techniken in diesen Sprachen (Vorlesung Programmierung); Kenntnisse aus der Theorie der Programmierung (z.B. Semantik und Verifikation).
Literatur	<ul style="list-style-type: none"> <li>Flemming Nielson, Hanne R. Nielson, Chris Hankin: Principles of Program Analysis. 2. Ausgabe, Springer, 2005</li> <li>Helmut Seidl, Reinhard Wilhelm, Sebastian Hack: Übersetzerbau 3: Analyse und Transformation. Springer, 2009</li> <li>Steven S. Muchnick, Neil D. Jones: Program Flow Analysis: Theory and Applications. Prentice Hall, 1981</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:  Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:  Dr. rer. nat. Katja Petzoldt  Modulverantwortlicher:  apl. Professor Dr. rer. nat. Thomas Noll</p>
ECTS Credits	6
Kontaktzeit (SWS)	5



- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Statische Programmanalyse (1212330)

Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Statische Programmanalyse (121233002)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Statische Programmanalyse (121233001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Statische Programmanalyse	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Stochastic Games (1219623)

Modultitel	Stochastic Games (Wahlpflichtfach)
Kennung	1219623
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Die Vorlesung beschäftigt sich mit dem Modell der stochastischen Spiele. In einem solchen Spiel wählen die Spieler in jeder Runde unabhängig voneinander ihre Aktionen, und der nächste Zustand des Spiels wird dann durch eine Wahrscheinlichkeitsverteilung bestimmt, die von den gewählten Aktionen abhängt.</p> <p>Es werden grundlegende Resultate und Algorithmen für solche Spiele mit unterschiedlichen Arten von Gewinnbedingungen vorgestellt.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>- Das Modell der stochastischen Spiele und deren grundlegende Eigenschaften</li> <li>- Algorithmische Verfahren zur Lösung stochastischer Spiele</li> <li>- Mathematische Methoden im Umgang mit stochastischen Spielen</li> </ul> <p>Kompetenzen:</p> <p>Die Studierenden sind in der Lage, stochastische Spiele als Modell zu verwenden. Sie beherrschen die wesentlichen mathematischen und algorithmischen Methoden im Umgang mit stochastischen Spielen und können diese anwenden.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse aus den Bereichen Automatentheorie und Wahrscheinlichkeitstheorie.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Priv.-Doz. Dr. rer. nat. Christof Löding
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Stochastic Games (1219623)

Selbststudium (h)	105,0
-------------------	-------

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Stochastic Games (121962302)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Stochastic Games (121962301)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Stochastik Games	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + The Graph Isomorphism Problem (1212654)

Modultitel	The Graph Isomorphism Problem (Wahlpflichtfach)
Kennung	1212654
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Das Graphenisomorphieproblem ist eines der wichtigsten offenen Probleme der theoretischen Informatik. Im Laufe der vergangenen fünfzig Jahre hat es eine Fülle von Teilergebnissen ganz unterschiedlicher Natur gegeben, die auf Techniken aus verschiedenen Teilgebieten der theoretischen Informatik und der Mathematik beruhen. Inhalt der Vorlesung sind die Höhepunkte der Forschung zum Graphenisomorphieproblem, angefangen mit frühen Ergebnissen aus den 1970er Jahren bis hin zu aktuellen Ergebnissen. Jedes dieser Ergebnisse wird eingebettet in eine Einführung in die verwendeten Techniken und den jeweiligen Kontext.</p>
Lernziele/Lernergebnisse	<p><b>Kenntnisse:</b></p> <ul style="list-style-type: none"> <li>- Kenntnisse im Forschungsbereich Graphenisomorphieproblem und die verwendeten Techniken.</li> </ul> <p><b>Fähigkeiten:</b></p> <ul style="list-style-type: none"> <li>- Erlernen fortgeschrittener Techniken aus verschiedenen Bereichen der theoretischen Informatik (Algorithmik, Komplexitätstheorie)</li> <li>- Erlernen fortgeschrittener Techniken aus angrenzenden Bereichen der Mathematik (Graphentheorie, algorithmische Gruppentheorie)</li> </ul> <p><b>Kompetenzen:</b></p> <ul style="list-style-type: none"> <li>- Kombination und Anwendung dieser Techniken im Kontext eines aktuellen Forschungsthemas.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Mathematische Grundlagen in den Bereichen Diskrete Strukturen und Lineare Algebra. Informatische Grundlagen in den Bereichen Datenstrukturen und Algorithmen, Berechenbarkeit und Komplexität.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:  Modulangebotsverantwortlicher Informatikmodellierungsteamverantwortlicher:  Dr. rer. nat. Katja Petzoldt  Modulverantwortlicher:  Universitätsprofessor Dr. rer. nat. Martin Grohe</p>

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + The Graph Isomorphism Problem (1212654)

ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exercise the Graph Isomorphism Problem (121265402)	1. Semester	keine Semesterempfehlung	0	2
Exam the Graph Isomorphism Problem (121265401)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture the Graph Isomorphism Problem	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Theory of Constraint Satisfaction Problems (1212334)

Modultitel	Theory of Constraint Satisfaction Problems (Wahlpflichtfach)
Kennung	1212334
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Constraint-Satisfaction Probleme (CSPs) bilden einen konzeptionellen Rahmen für eine Vielzahl algorithmischer Probleme mit Anwendungen in verschiedenen Bereichen der Informatik. Inhalt der Vorlesung sind Algorithmen zur Lösung von CSPs sowie die komplexitätstheoretische Klassifikation von CSPs.
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>- Algorithmen für CSPs</li> <li>- Komplexitätstheoretische Klassifikation von CSPs</li> </ul> <p>Fähigkeiten:</p> <ul style="list-style-type: none"> <li>- Erlernen fortgeschrittener Techniken aus verschiedenen Bereichen der theoretischen Informatik (Algorithmik, Logik, Komplexitätstheorie)</li> <li>- Erlernen fortgeschrittener Techniken aus angrenzenden Bereichen der Mathematik (diskrete Mathematik, universelle Algebra)</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>- Kombination und Anwendung dieser Techniken im Kontext eines aktuellen Forschungsthemas.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Mathematische Grundlagen den Bereichen Diskrete Strukturen und Lineare Algebra. Informatische Grundlagen in den Bereichen Datenstrukturen und Algorithmen, Berechenbarkeit und Komplexität sowie Logik.
Literatur	-
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher Informatikmodellierungsteamverantwortlicher:</p> <p>Dr. rer. nat. Katja Petzoldt</p> <p>Modulverantwortlicher:</p> <p>Universitätsprofessor Dr. rer. nat. Martin Grohe</p>
ECTS Credits	6

- Theoretical Foundations of Software ...
- Wahlpflichtbereich Theoretische ...
- + Theory of Constraint Satisfaction Problems (1212334)

Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exercise Theory of Constraint Satisfaction Problems (121233402)	1. Semester	keine Semesterempfehlung	0	2
Exam Theory of Constraint Satisfaction Problems (121233401)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Theory of Constraint Satisfaction Problems	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Algorithmic Foundations of Datascience (1216860)

Modultitel	Algorithmic Foundations of Datascience (Wahlpflichtfach)
Kennung	1216860
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Die Vorlesung behandelt ein breites Spektrum von algorithmischen Techniken aus dem Bereich Data Science. Der Schwerpunkt liegt dabei auf mathematischen und theoretischen Aspekten wie einer sorgfältigen Komplexitätsanalyse. Ausgehend von den mathematischen Grundlagen aus Stochastik, Informationstheorie und linearer Algebra werden typische Datenanalysealgorithmen sowie typische Algorithmen zum Verarbeiten großer Datenmengen etwa auf Computerclustern oder in Datenstromszenarien behandelt.
Lernziele/Lernergebnisse	<p><b>Kenntnisse:</b> Ein breites Spektrum verschiedener Data-Science Algorithmen und ihre mathematischen Grundlagen.</p> <p><b>Fähigkeiten:</b> Entwicklung eines theoretischen Verständnisses für algorithmische Effizienz und Komplexität in Data-Science Szenarios. Techniken zur Analyse von Algorithmen in diesem Bereich.</p> <p><b>Kompetenzen:</b> Kompetenz zur Auswahl geeigneter Algorithmen in verschiedenen Anwendungsszenarien und ein Verständnis für die Implikationen dieser Auswahl.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Mathematische Grundlagen den Bereichen Lineare Algebra und Wahrscheinlichkeitstheorie. Informatische Grundlagen in den Bereichen Datenstrukturen und Algorithmen, Berechenbarkeit und Komplexität, sowie Datenbanksysteme.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5



- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Algorithmic Foundations of Datascience (1216860)

Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Algorithmic Foundation of Data Science (121686002)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Algorithmic Foundation of Data Science (121686001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Algorithmic Foundations of Data Science	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Erfüllbarkeitsüberprüfung (1212341)

Modultitel	Erfüllbarkeitsüberprüfung (Wahlpflichtfach)
Kennung	1212341
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>- Propositional logic, the satisfiability problem</li> <li>- SAT-solving for checking the satisfiability of propositional logic formulas</li> <li>- First-order logic, theories</li> <li>- Satisfiability modulo theories (SMT) solving</li> <li>- SMT solving for the theory of equalities and uninterpreted functions</li> <li>- SMT solving for bit-vector arithmetic</li> <li>- SMT solving for linear real arithmetic</li> <li>- SMT solving for linear integer arithmetic</li> <li>- SMT solving for (non-linear) real arithmetic</li> <li>- Applications</li> </ul>
Lernziele/Lernergebnisse	<p><b>Knowledge:</b> In this lecture the students will learn to differentiate between different first-order theories and get insight into corresponding decidability and complexity results. The lecture will show them how satisfiability checking algorithms check formulas from those logics for satisfiability.</p> <p><b>Skills:</b> The students will practice to formalize problems in adequate logics and to apply satisfiability checking procedures to solve them. Especially, they will know which solvers are available and can use these e.g. for verification and counterexample generation.</p> <p><b>Competences:</b> The students will improve their competences for the development and application of decision procedures. They will be able to decide when they can use these methods to solve problems from different areas of computer science.</p> <p>In the lecture they will improve the exact communication of scientific problems. The lecture will also increase their motivation for the application of formal methods.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Knowledge from the modules "Mathematical logic", as well as "Algorithms and data structures".
Literatur	<p>Folien der Vorlesung und die folgenden Bücher:</p> <ul style="list-style-type: none"> <li>• Daniel Kroening, Ofer Strichman: Decision Procedures: An Algorithmic Point of View. Springer Berlin, 2008</li> <li>• Aaron R. Bradley, Zohar Manna: The Calculus of Computation: Decision Procedures with Applications to Verification. Springer, Berlin. 2007</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessorin Dr. Erika Ábrahám

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Erfüllbarkeitsüberprüfung (1212341)

ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Erfüllbarkeitsüberprüfung (121234102)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Erfüllbarkeitsüberprüfung (121234101)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Erfüllbarkeitsüberprüfung	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Formale Grundlagen von UML (1212648)

Modultitel	Formale Grundlagen von UML (Wahlpflichtfach)
Kennung	1212648
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Main topics:</p> <ul style="list-style-type: none"> <li>• Sequence diagrams and their semantics</li> <li>• Elementary properties of sequence diagrams</li> <li>• High-level sequence graphs</li> <li>• Communicating finite automata</li> <li>• Realizability</li> <li>• Statecharts and their semantics</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge</p> <p>on completion of this course, students have acquired basic knowledge about</p> <ul style="list-style-type: none"> <li>• Message sequence charts</li> <li>• High-level sequence graphs</li> <li>• Propositional dynamic logic</li> <li>• Statecharts</li> <li>• Communicating finite-state automata</li> <li>• Realisability</li> </ul> <p>Skills</p> <p>on completion of this course, students are skilled to</p> <ul style="list-style-type: none"> <li>• Model software systems in a component-based way</li> <li>• Analyse the precise meaning of software system models and specifications</li> </ul> <p>Competences</p> <p>on completion of this course, students are able to</p> <ul style="list-style-type: none"> <li>• Apply formal modelling techniques to software systems</li> <li>• Read, understand, and write formal system requirement specifications</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Knowledge of fundamental automata models such as finite and pushdown automata. Fundamental knowledge of mathematical logic. Knowledge of discrete mathematics. Basic knowledge of complexity theory.
Literatur	<p>Lecture slides and script as well as:</p> <ul style="list-style-type: none"> <li>• D. Harel and M. Politi, Modeling Reactive Systems with Statecharts: The STATEMATE Approach, McGraw-Hill, 1998.</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Formale Grundlagen von UML (1212648)

Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Formale Grundlagen von UML (121264802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Formale Grundlagen von UML (121264801)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Formale Grundlagen von UML	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Formale Methoden für Steuerungssoftware (1212666)

Modultitel	Formale Methoden für Steuerungssoftware (Wahlpflichtfach)
Kennung	1212666
Version	V2
Dauer (Semester)	-
Turnus (Semester)	-
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	-
Inhalt	<p>Die Vorlesung wird eine anwendungsorientierte Einführung in verschiedene Formale Methoden geben. Dabei liegt der Fokus auf dem Einsatz dieser bei der Entwicklung von Steuerungssoftware. Die Themen der Vorlesung umfassen:</p> <ul style="list-style-type: none"> <li>• Statische Analyse</li> <li>• Abstract Interpretation</li> <li>• Program Slicing</li> <li>• Spezifikationen</li> <li>• Model-Checking</li> <li>• Concolic Testing</li> </ul>
Lernziele/Lernergebnisse	<p>Wissen: Nach Abschluss des Moduls werden die Studierenden die folgenden Punkte erklären können:</p> <ul style="list-style-type: none"> <li>• Herausforderungen bei der Analyse von Steuerungssoftware</li> <li>• Techniken und Formalismen zur statischen Analyse</li> <li>• Vor- und Nachteile verschiedener abstrakter Domänen</li> <li>• Methoden und Anwendungen von Program Slicing</li> <li>• Modellierung von SPS-Programmen zwecks Verifikation</li> <li>• Verschiedene Techniken zur Spezifikation von Systemanforderungen</li> <li>• Verfahren zur Verifikation verschiedener Spezifikationen</li> <li>• Techniken zur Kodierung von SPS-Semantik als Formeln für SMT-Solver</li> </ul> <p>Fähigkeiten: Die Studierenden sollten in der Lage sein, die folgenden Techniken zu erklären und anzuwenden:</p> <ul style="list-style-type: none"> <li>• Aufbau eines Frameworks für statische Analyse</li> <li>• Ableiten verschiedener Programmeigenschaften durch die Wahl geeigneter Abstrakter Domänen für die statische Analyse</li> <li>• Abhängigkeitsanalyse und Vereinfachung durch Program Slicing</li> <li>• Kodierung von SPS-Programmen als Formeln zur Erfüllbarkeitsüberprüfung mit SMT-Solvern</li> <li>• Spezifikationen von Anforderungen durch geeignete Formalisierungen entwerfen</li> <li>• Überprüfen von Spezifikationen mittels geeigneter Model-Checking Algorithmen</li> </ul> <p>Kompetenzen: Basierend auf dem gewonnenen Wissen und Fähigkeiten sollten die Studierenden in der Lage sein</p> <ul style="list-style-type: none"> <li>• Eine statische Analyse für Steuerungssoftware mit geeigneten abstrakten Domänen zu entwickeln,</li> <li>• Programme und Anforderungen zu modellieren, bzw. spezifizieren und</li> <li>• Geeignete Techniken aus einem Fundus Formaler Methoden auf Problemstellungen im Bereich der Programmanalyse anzuwenden.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Vorkenntnisse in statischer Analyse und/oder Modellüberprüfung sind von Vorteil, aber nicht zwingend erforderlich.
Literatur	-
Sprache	Englisch

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Formale Methoden für Steuerungssoftware (1212666)

Prüfungsbedingungen	Klausur (100 %).
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Universitätsprofessor Dr.-Ing. Stefan Kowalewski Hendrik Simon M. Sc. RWTH
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Formale Methoden für Steuerungssoftware (121266602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Formale Methoden für Steuerungssoftware (121266601)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Formale Methoden für Steuerungssoftware	1. Semester	keine Semesterempfehlung	-	2

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Foundations of Functional Programming (1215684)

Modultitel	Foundations of Functional Programming (Wahlpflichtfach)
Kennung	1215684
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Einführung in die Programmiersprache Haskell</p> <ul style="list-style-type: none"> <li>• Syntax der verschiedenen Sprachkonstrukte</li> <li>• Funktionen höherer Ordnung</li> <li>• Programmieren mit Lazy Evaluation</li> </ul> <p>Denotationelle Semantik funktionaler Programme</p> <ul style="list-style-type: none"> <li>• Vollständige Ordnungen und Fixpunkte</li> <li>• Denotationelle Semantik von Haskell</li> </ul> <p>Der Lambda-Kalkül</p> <ul style="list-style-type: none"> <li>• Syntax und operationelle Semantik des Lambda-Kalküls</li> <li>• Reduzierung von Haskell auf den Lambda-Kalkül</li> </ul> <p>Typ-Überprüfung und Typ-Inferenz</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Kenntnis der Konzepte, die funktionalen Programmiersprachen zugrunde liegen</p> <p>Fähigkeiten:</p> <ul style="list-style-type: none"> <li>• Erlernen der Programmier Techniken in funktionalen Sprachen</li> <li>• Fähigkeit zur formalen Festlegung der Semantik funktionaler Programmiersprachen</li> <li>• Fähigkeit zur Implementierung funktionaler Sprachen</li> <li>• Fähigkeit zum Entwurf von Verfahren zur Typüberprüfung bei funktionalen Sprachen</li> </ul> <p>Kompetenzen: Erlernen, wie man funktionale Sprachen in verschiedenen Anwendungsgebieten einsetzen kann</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Basic programming concepts. Prior knowledge on functional programming would be advantageous, but is not required.
Literatur	<p>Skript und Folien zur Vorlesung sowie z.B. folgende Bücher:</p> <ul style="list-style-type: none"> <li>• R. Bird: Thinking Functionally With Haskell, Cambridge University Press, 2014.</li> <li>• G. Hutton: Programming in Haskell, Cambridge University Press, 2016.</li> <li>• B. O'Sullivan, D. Stewart, J. Goerzen: Real World Haskell, O'Reilly, 2010.</li> <li>• P. Pepper: Funktionale Programmierung, Springer, 2002.</li> <li>• C. Reade: Elements of Functional Programming, Addison-Wesley, 1989.</li> <li>• P. Thiemann: Grundlagen der Funktionalen Programmierung, Teubner, 1994.</li> </ul>
Sprache	Englisch



- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Foundations of Functional Programming (1215684)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Jürgen Giesl
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exercise Foundations of Functional Programming (121568402)	1. Semester	keine Semesterempfehlung	0	2
Exam Foundations of Functional Programming (121568401)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Foundations of Functional Programming	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Foundations of Logic Programming (1212343)

Modultitel	Foundations of Logic Programming (Wahlpflichtfach)
Kennung	1212343
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Prädikatenlogische Grundlagen</p> <ul style="list-style-type: none"> <li>• Unifikation</li> <li>• Resolution</li> <li>• Horn-Klauseln und SLD-Resolution</li> </ul> <p>Logikprogramme</p> <ul style="list-style-type: none"> <li>• Operationelle und denotationelle Semantik</li> <li>• Auswertungsstrategien</li> </ul> <p>Die Programmiersprache Prolog</p> <ul style="list-style-type: none"> <li>• Negation as Failure</li> <li>• Nicht-logische Bestandteile von Prolog</li> <li>• Programmiertechniken</li> </ul> <p>Anwendungen und Erweiterungen der Logikprogrammierung</p>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <p>Kenntnis der Konzepte, die logischen Programmiersprachen zugrunde liegen</p> <p>Fähigkeiten:</p> <ul style="list-style-type: none"> <li>• Erlernen der Programmiertechniken in logischen Programmiersprachen</li> <li>• Fähigkeit zur formalen Festlegung der Semantik logischer Programmiersprachen</li> <li>• Fähigkeit zur Implementierung logischer Sprachen</li> </ul> <p>Kompetenzen:</p> <p>Erlernen, wie man logische Sprachen in verschiedenen Anwendungsgebieten einsetzen kann</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Basic programming concepts. Prior knowledge on logic programming would be advantageous, but is not required. Prior knowledge on predicate logic would be advantageous, but is not required.
Literatur	<p>Skript und Folien zur Vorlesung sowie z.B. folgende Bücher:</p> <ul style="list-style-type: none"> <li>• I. Bratko: Prolog Programming for Artificial Intelligence, Addison-Wesley, 2011.</li> <li>• W. F. Clocksin, C. S. Mellish: Programming in Prolog, Springer, 2013.</li> <li>• T. Frühwirth, S. Abdennadher: Essentials of Constraint Programming, Springer, 2010.</li> <li>• M. Hanus: Problemlösen mit Prolog, Teubner, 1987.</li> <li>• J. W. Lloyd: Foundations of Logic Programming, Springer, 2013.</li> <li>• P. H. Schmitt: Theorie der logischen Programmierung, Springer, 1992.</li> <li>• L. Sterling, E. Shapiro: The art of Prolog, MIT Press, 2000.</li> </ul>
Sprache	Englisch

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Foundations of Logic Programming (1212343)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Jürgen Giesl
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exercise Foundations of Logic Programming (121234302)	1. Semester	keine Semesterempfehlung	0	2
Exam Foundations of Logic Programming (121234301)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Foundations of Logic Programming	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Model Checking (1212328)

Modultitel	Model Checking (Wahlpflichtfach)
Kennung	1212328
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Main topics:</p> <ul style="list-style-type: none"> <li>• Transition systems</li> <li>• Concurrent and channel systems</li> <li>• Property classes: safety, liveness, invariants, and fairness</li> <li>• Linear Temporal Logic (LTL)</li> <li>• Computation Tree Logic (CTL) Model Checking algorithms for LTL and (fair) CTL</li> <li>• Abstraction: (Bi)simulation</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge</p> <p>on completion of this course, students have acquired detailed knowledge about</p> <ul style="list-style-type: none"> <li>• Modelling of concurrent programs</li> <li>• Elementary property classes: safety, liveness, and fairness</li> <li>• Verification algorithms for automata on finite and infinite words</li> <li>• Model-checking algorithms for temporal logics LTL and CTL</li> <li>• Expressiveness of LTL versus CTL</li> </ul> <p>Skill</p> <p>on completion of this course, students are skilled to</p> <ul style="list-style-type: none"> <li>• Solving of moderately-sized model-checking problems</li> <li>• Reasoning with and using temporal logic</li> </ul> <p>Competences</p> <p>on completion of this course, students are able to</p> <ul style="list-style-type: none"> <li>• Judge the applicability of model checking to practical cases.</li> <li>• Model concurrent programs and formulate their basic properties in temporal logic</li> <li>• Apply model-checking algorithms to small transition systems</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Knowledge of fundamental automata models and regular languages. Knowledge of propositional logic. Knowledge of basic data structures such as stacks, trees, and graphs and related algorithms. Basic knowledge of complexity theory.
Literatur	<p>Folien zur Vorlesung sowie folgende Lehrbücher:</p> <ul style="list-style-type: none"> <li>• C. Baier, J.-P. Katoen: Principles of Model Checking, MIT Press, 2008.</li> <li>• M. Huth and M.D. Ryan: Logic in Computer Science, Modelling and Reasoning about Systems, Cambridge Univ. Press, 2004.</li> <li>• E.M. Clarke, O. Grumberg, D. Peled: Model Checking, MIT Press, 1999.</li> </ul>
Sprache	Englisch

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Model Checking (1212328)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor i.R. Dr. rer. nat. Dr. h. c. Dr. h. c. Wolfgang ThomasUniversitätsprofessor Dr. ir. Dr. h. c. (AAU) Joost-Pieter Katoen
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Model Checking (121232802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Model Checking (121232801)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Model Checking	1. Semester	keine Semesterempfehlung	-	3

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Theory of Distributed and Parallel Systems (1212643)

Modultitel	Theory of Distributed and Parallel Systems (Wahlpflichtfach)
Kennung	1212643
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Introduction to distributed and parallel systems, different models</li> <li>• Algorithms in message-passing systems</li> <li>• Leader election and consensus</li> <li>• Distributed Hash Tables for Peer-2-Peer Networks (Chord)</li> <li>• Algorithms in parallel systems</li> <li>• Complexity (P-Completeness)</li> <li>• Routing and communication in networks</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: At the end of this course, the students should be able to describe foundational problems arising in distributed and parallel systems and explain algorithmic solutions for these problems. Skills: The students should be able to apply general algorithmic design principles and to use techniques like load balancing and randomization to solve problems arising in network and parallel contexts. Competences: The students should be able to model distributed and parallel systems in a formal way and to develop algorithmic solutions enabling the efficient usage of computer networks and other distributed systems.</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Basic knowledge about algorithms, discrete structures, and probability theory.
Literatur	<ul style="list-style-type: none"> <li>• Leighton. Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes</li> <li>• Kurose, Ross: Computer Networking: A Top-Down Approach Featuring the Internet. Addison Wesley Longman, 1999.</li> <li>• Kleinberg, Tardos: Algorithm Design, Addison Wesley Pearson, 2005</li> <li>• Gibbons, Rytter: Efficient Parallel Algorithms, Cambridge University Press</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	This module can alternatively be completed as 6 ECTS (3 hours of lecture and 2 hours of exercises, V3+Ü2) 8 ECTS (3+1 hours of lecture and 2 hours of exercises, V4+Ü2). Written exam or oral examination (100 %). Students must pass written homework to be admitted to the module examination.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator: Modulangebotsverantwortlicher Informatikmodellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Dr. rer. nat. Walter Unger</p>
ECTS Credits	6
Kontaktzeit (SWS)	6

- Theoretical Foundations of Software ...
- Core Subjects Theoretical Foundations of ...
- + Theory of Distributed and Parallel Systems (1212643)

Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Theory of Distributed and Parallel Systems (121264303)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Theory of Distributed and Parallel Systems (121264302)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Theory of Distributed and Parallel Systems	1. Semester	keine Semesterempfehlung	-	3
Erweiterte Vorlesung Theory of Distributed and Parallel Systems	1. Semester	keine Semesterempfehlung	-	1

Modultitel	Ad-Hoc Networks: Architectures and Protocols (Wahlpflichtfach)
Kennung	6010396
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2010
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Principles of ad hoc, heterogeneous, and local area wireless networks</li> <li>• Connectivity and routing in wireless networks</li> <li>• MAC schemes for wireless networks</li> <li>• Energy efficiency of wireless protocols</li> <li>• Wireless network design and standards for modern and emerging applications, e.g. UHD multimedia, IoT, M2M, Industry 4.0, V2X</li> </ul>
Lernziele/Lernergebnisse	<p>Specific skills:</p> <ul style="list-style-type: none"> <li>• Students will understand the fundamental design principles of medium access control and network protocols for ad hoc, heterogeneous and local area wireless networks</li> <li>• Students will be able to describe the functionalities of common wireless medium access control and network protocols and to put them into the context of system design requirements</li> <li>• Students will be able to reason about the design space of feasible wireless communication technologies and network designs for modern and emerging application areas, e.g. broadband multimedia multimedia, IoT, M2M, Industry 4.0, V2X, etc.</li> </ul> <p>General skills:</p> <ul style="list-style-type: none"> <li>• Students will broaden their knowledge of multi-parametric systems design</li> <li>• Students will extend their knowledge on mathematical ;; modelling ;; through abstraction</li> <li>• Students learn to navigate the relevant scientific Reading Liste and review it critically</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basic knowledge in the design of communication networks
Literatur	<ul style="list-style-type: none"> <li>• C. Siva Ram Murthy and B.S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Prentice Hall</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	oral examination (30min) or written examination (90min)
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Petri Mähönen
ECTS Credits	4



## — Communication

## + Ad-Hoc Networks: Architectures and Protocols (6010396)

Kontaktzeit (SWS)	3
Prüfungsdauer (min)	90 or 30
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam Ad-Hoc Networks: Architectures and Protocols (601039601)	2. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture and Exercise Ad-Hoc Networks: Architectures and Protocols	2. Semester	keine Semesterempfehlung	-	3

Modultitel	Advanced Internet Technology (Wahlpflichtfach)
Kennung	1215688
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Einführende Veranstaltungen in Kommunikationssysteme behandeln die klassischen Prinzipien und Protokolle des Internets. Diese werden zwar immer noch verwendet, genügen jedoch oft den Anforderungen moderner Netzwerke nicht mehr. Dieser Kurs behandelt aufbauend auf den klassischen Prinzipien neuere Entwicklungen der Internet-Technologie:</p> <ul style="list-style-type: none"> <li>• Realisierung skalierbarer Anwendungen und Kommunikation: Peer-to-Peer-Systeme und Cloud Computing</li> <li>• Integration Ressourcen-beschränkter Geräte in das Internet: Cyber-physical Systems und das Internet of Things</li> <li>• Realisierung adaptiver Kommunikation: Software Defined Networking und Quality of Service</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Verständnis für die Beschränkungen klassischer Kommunikationsprinzipien im heutigen Internet</li> <li>• Kenntnis der Grundprinzipien zur Umsetzung skalierbarer, adaptiver und Ressourcen-beschränkter Kommunikation</li> </ul> <p>Fertigkeiten:</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur Identifikation von Problemen klassischer Kommunikationsprotokolle in modernen Kommunikationssystemen</li> <li>• Fähigkeit zur Anwendung der Algorithmen hinter skalierbarer, adaptiver und Ressourcen-beschränkter Kommunikation</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur methodischen Analyse der Anwendbarkeit von Lösungen zur skalierbaren, adaptiven und Ressourcen-beschränkten Kommunikation auf zukünftige Internetszenarien</li> <li>• Fähigkeit zur Identifikation von Weiterentwicklungsmöglichkeiten moderner Kommunikationssysteme</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Inhalte der Vorlesung Data Communication and Security sind hilfreich.
Literatur	<ul style="list-style-type: none"> <li>• Vorlesungsfolien / Lecture Slides</li> <li>• Steinmetz, Wehrle (Eds.): Peer-to-Peer Systems and Applications, Springer, 2005</li> <li>• Karl, Willig: Protocols and Architectures for Wireless Sensor Networks, Wiley, 2005</li> <li>• Weitere Spezialliteratur wird in den Vorlesungsfolien bekannt gegeben</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.

— Communication

+ Advanced Internet Technology (1215688)

Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Klaus Wehrle
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Advanced Internet Technology (121568802)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Advanced Internet Technology (121568801)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Advanced Internet Technology	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Advanced Network Security (Wahlpflichtfach)
Kennung	1230527
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	In diesem Modul werden kryptographische Protokolle und Sicherheitsarchitekturen von drahtlosen und drahtgebundenen Netzen untersucht. Ein besonderer Schwerpunkt wird dabei auf kryptographischen Schwächen und Angriffen liegen. Betrachtet werden die Architekturen von und Angriffe auf Mobilfunknetze verschiedener Generationen, WLAN, Bluetooth, EAP und EAP-Methoden, BGP, FIDO2, sowie bereits bekannte Protokolle wie TLS.
Lernziele/Lernergebnisse	<p>Kenntnisse: Nach erfolgreichem Absolvieren des Moduls können die Studierenden die Sicherheitsarchitekturen gängiger Kommunikationsnetze erläutern, die Sicherheitseigenschaften verschiedener Netze miteinander vergleichen und gängige Designschwächen und Angriffe auf Sicherheitsarchitekturen und Sicherheitsprotokolle erläutern sowie Lösungsansätze zu deren Behebung beschreiben.</p> <p>Fertigkeiten: Sicherheitsanforderungen und Schwächen von Netzen und Protokollen analysieren und die anzuwendenden Sicherheitsmechanismen bzw. Lösungsansätze zu deren Behebung bestimmen.</p> <p>Kompetenzen: Basierend auf den im Modul erworbenen Kenntnisse und Fertigkeiten, können Studierende die Übertragung wesentlicher bekannter Angriffsstrategien auf existierende und neue Sicherheitsarchitekturen und Protokollen wissenschaftlich diskutieren und eigenständig Empfehlung zu deren Schutz entwickeln.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Das Modul "Sicherheit in der Mobilkommunikation" (1212681) darf nicht erfolgreich abgeschlossen sein.
(empfohlene) Voraussetzungen	<p>Grundkenntnisse in Kryptographie und Netzwerksicherheit entsprechend der Vorlesung IT-Sicherheit aus dem Bachelor Informatik.</p> <p>Die Vorlesung ergänzt sich sehr gut mit der Vorlesung Mobile Internet Technology. Beide Vorlesungen können in beliebiger Reihenfolge gehört werden, auch parallel.</p>
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. Dr.-Ing. Ulrike Meyer
ECTS Credits	6
Kontaktzeit (SWS)	4

— Communication

+ Advanced Network Security (1230527)

Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Advanced Network Security (123052701)	1. Semester	keine Semesterempfehlung	6	0
Übung Advanced Network Security (123052702)	1. Semester	keine Semesterempfehlung	0	1

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Advanced Network Security	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Communication Systems Engineering (Wahlpflichtfach)
Kennung	1212349
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Diese Vorlesung vermittelt die Grundlagen und Technologien des Engineerings moderner Kommunikationssysteme:</p> <ul style="list-style-type: none"> <li>• Protokoll-Design: grundlegende Methoden des Designs von Kommunikationsprotokollen</li> <li>• Implementierung: Werkzeuge und Technologien zur Implementierung von Kommunikationssystemen sowohl im Kernel-Space als auch im User-Space</li> <li>• Verifikation und Testen: Ansätze zur Sicherstellung korrekten Verhaltens einer Protokollimplementierung</li> <li>• Evaluation durch diskrete, ereignisorientierte Simulation (Modellierung, Validierung, Parameterstudien)</li> <li>• Internet-weite Evaluation durch Messungen.</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Verständnis für die Grundlagen des Protokoll-Engineerings</li> <li>• Kenntnis unterschiedlicher Konzepte zur Leistungsbewertung</li> </ul> <p>Fertigkeiten:</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur Analyse der Ergebnisse einer Leistungsbewertungsstudie</li> <li>• Fähigkeit zur Identifikation von Verbesserungsmöglichkeiten von Kommunikationssystemen</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur Konzeption neuartiger Kommunikationssysteme und der Bewertung ihrer Leistungsfähigkeit.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Inhalte der Vorlesungen Datenkommunikation und Sicherheit sowie Betriebssysteme und Systemsoftware. Kenntnisse in C/C++-Programmierung sind empfehlenswert.
Literatur	<ul style="list-style-type: none"> <li>• Vorlesungsfolien / Lecture Slides</li> <li>• Auf weitere Literatur wird in den Folien verwiesen.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Klaus Wehrle & Dr. rer. nat. Dirk Thißen
ECTS Credits	6

— Communication

+ Communication Systems Engineering (1212349)

Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Communication Systems Engineering (121234901)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung/Übung Communication Systems Engineering	1. Semester	keine Semesterempfehlung	-	4

Modultitel	Dynamical Processes on Networks (Wahlpflichtfach)
Kennung	1223640
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2021
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Many real-world systems may be described as a network of dynamically interacting entities. We interact with each other in a social contact network, over which rumours as well as pathogens can spread; electrical energy is delivered by the power grid; the Internet enables almost instantaneous world-wide interactions; our economies rest upon a complex network of inter-dependencies spanning the globe. Networks are ubiquitous in complex biological, social, engineering, and physical systems. Understanding the dynamics of these systems is essential if we are to redesign them, or guide/control them towards different behaviours. Networked control problems abound and include multi-user communication, distributed computation and sensing, swarming, flocking, and synchronization.</p> <p>This course provides a (mathematical) introduction to such network dynamical systems. We discuss a selection of fundamental dynamical phenomena over interconnected network systems, e.g., consensus and disagreement in averaging systems, epidemic spreading dynamics, opinion formation models and synchronization of coupled oscillators and networked control systems.</p>
Lernziele/Lernergebnisse	<p>This course will provide students with the mathematical tools and computational training to understand large-scale dynamical networks.</p> <p>Knowledge</p> <ul style="list-style-type: none"> <li>Students will be familiar with typically considered distributed dynamical systems on networks.</li> <li>These include linear processes such as consensus dynamics &amp; distributed averaging, network diffusion models &amp; opinion formation models, and nonlinear processes such as epidemic spreading (SI, SIS, SIR) and synchronization processes.</li> </ul> <p>Skills</p> <ul style="list-style-type: none"> <li>After taking this course the students will be able to model a variety of systems as networks.</li> <li>Students will be able to analyse key features of the dynamics of such dynamical network systems, using algebraic graph theory and relate (structural) properties of the network to observed dynamical behaviours. They will further be able to simulate and analyse the dynamics of such systems using python code.</li> </ul> <p>Competences</p> <ul style="list-style-type: none"> <li>Students will be able to use networks as a language to communicate across different application domains and be able to analyze and implement simple models for network dynamics across a variety of domains.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	<p>Knowledge of Linear Algebra  Knowledge of basic Probability  Knowledge of basic Graph Theory  Knowledge of Python  (optional) Some prior experience with dynamical systems</p>



## — Communication

## + Dynamical Processes on Networks (1223640)

Literatur	F. Bullo, "Lectures on Network Systems", 2020, available online at <a href="http://motion.me.ucsb.edu/book-Ins/">http://motion.me.ucsb.edu/book-Ins/</a> Additional material will be provided in the form of slides and research papers.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Klausur Dynamical Processes on Networks (122364001)	1. Semester	keine Semesterempfehlung	6	0
Übung Dynamical Processes on Networks (122364002)	1. Semester	keine Semesterempfehlung	0	2

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Dynamical Processes on Networks	keine Semesterempfehlung	keine Semesterempfehlung	-	3

Modultitel	Industrial Network Security (Wahlpflichtfach)
Kennung	1227956
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2022
Gültig bis	-
Modulniveau	Master
Inhalt	This course provides an introduction into current security problems of industrial networks and state-of-the-art security solutions. After an introduction into industrial networks and protocols and their prevalent security issues, students will learn about approaches to prevent and detect as well as react to cyberattacks and security incidents. Topics include security approaches for legacy industrial control systems with tight resource-constraints, security assessments of industrial networks, and intrusion detection and prevention in future industrial networks.
Lernziele/Lernergebnisse	<p>This course will provide students with an understanding of the unique challenges of realizing security within industrial networks and provide them with the necessary conceptual tools to properly secure industrial networks. ;</p> <p><b>Knowledge:</b> Students will be familiar with the security challenges of industrial networks, specifically focusing on technical constraints different than in traditional computer networks. Students will know about the various approaches to secure industrial networks, covering the prevention and detection of as well as the reaction to advanced cyberattacks against industrial networks.</p> <p><b>Skills:</b> After taking this course, students will be able to select and apply suitable conceptual tools to realize comprehensive security of industrial networks. Based on an assessment of the individual constraints in a given industrial scenario, students will be able to select those security measures that are applicable and prioritize their deployment.</p> <p><b>Competences:</b> Students will be able to identify and pinpoint security prevalent in industrial networks across different domains. They will further be able to identify where standard security approaches can be adapted to secure industrial networks and where dedicated security approaches are required.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Basic knowledge in the field of data communication and security, e.g., as taught in lectures such as "Data Communication and Security", "Data Communication", or "IT-Security".
Literatur	Lecture slides. Pointers to further relevant literature will be provided during the lecture.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Prof. Dr. Martin Henze

— Communication

+ Industrial Network Security (1227956)

ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Industrial Network Security (Übung) (122795602)	1. Semester	keine Semesterempfehlung	0	1
Industrial Network Security (122795601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Industrial Network Security (Vorlesung)	1. Semester	keine Semesterempfehlung	-	3

Modultitel	IT-Sicherheit 1 - Kryptographische Grundlagen und Netzwerksicherheit (Wahlpflichtfach)
Kennung	1211901
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	The lecture consists of two parts. The first part covers the cryptographic basics including: Symmetric Encryption, Integrity protection, Asymmetric Encryption, Digital Signatures, Certificates and Public Key Infrastructures, and Authentication and Key Agreement. The second part is dedicated to Network Security including Kerberos, IPsec, TLS protocol, SSH, DNS Security, Email Security, and Phishing Attacks.
Lernziele/Lernergebnisse	Knowledge: On successful completion of this module, students should be able to define the cryptographic primitives symmetric / asymmetric encryption, digital signatures, cryptographic hash function, and message authentication codes. They should be able to explain the security features offered by the latest versions of the most important security protocols operating on the TCP/IP stack (IPsec, TLS, SSH, DNSsec, PGP) and describe known attacks against these security protocols.   Skills: the students should be able to select and apply the appropriate cryptographic primitives in different application scenarios. They should be able to select the appropriate security protocols in a given scenario and configure the appropriate options for the selection of the appropriate cryptographic building blocks applied within the studied protocols.   Competences: Based on the knowledge and skills acquired they should be able to identify the security requirements and adequate security mechanisms in different areas of application. In addition they should be able to identify potential weaknesses in security protocols that have not been studied in detail within the lecture and be able to suggest fixes to the identified weaknesses. Finally they should be able to assess the severity of new attacks against security protocols and cryptographic primitives.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basics of Data Communication and Modular Arithmetic.
Literatur	<ul style="list-style-type: none"> <li>• Introduction to Cryptography and Network Security, Forouzan, Mc Graw-Hill International</li> <li>• Network Security: Private Communication in a Public World, Kaufmann, Perlman, and Speciner, Prentice Hall</li> <li>• Cryptography and Network Security - Principles and Practice, Stallings, Prentice Hall</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessorin Dr.-Ing. Ulrike Meyer
ECTS Credits	6
Kontaktzeit (SWS)	4

— Communication

+ IT-Sicherheit 1 - Kryptographische Grundlagen und ...

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung IT-Sicherheit 1 (121190102)	1. Semester	keine Semesterempfehlung	0	1
Prüfung IT-Sicherheit 1 (121190101)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung IT-Sicherheit 1	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Mobile Internet Technology (Wahlpflichtfach)
Kennung	1212346
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Dieser Kurs befasst sich mit Architekturen, Protokollen und Algorithmen für mobile Internet-Systeme:</p> <ul style="list-style-type: none"> <li>• Physical Layer: Modulation, Codierung und Signalausbreitung</li> <li>• MAC Layer: Herausforderungen beim Medienzugriff auf Funkkanälen</li> <li>• Drahtlose, datenorientierte Netze: 802.11 (WLAN)</li> <li>• Routing in Ad-hoc-Netzen</li> <li>• Mobilfunk: GSM, GPRS, UMTS, LTE, 5G</li> <li>• Mobilität im Internet: Mobile IP, HIP, TCP</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Kenntnis der Funktionsprinzipien drahtloser Netze, speziell 802.11 (WLAN) und Mobilfunk</li> <li>• Kenntnis der Probleme der Internetprotokolle (IP, TCP) in mobilen Szenarien</li> </ul> <p>Fertigkeiten:</p> <ul style="list-style-type: none"> <li>• Fähigkeit, Problemquellen in mobilen Szenarien zu identifizieren und geeignet zu lösen</li> <li>• Fähigkeit zur Identifikation wichtiger gemeinsamer Aspekte verschiedener drahtloser Netzwerklösungen</li> </ul> <p>Kompetenzen: ;</p> <ul style="list-style-type: none"> <li>• Fähigkeit zur methodischen Analyse der Anwendbarkeit von Architekturen drahtloser, mobiler Systeme auf zukünftige Internetszenarien</li> <li>• Fähigkeit zur Diskussion von Anforderungen an Internetprotokolle in drahtlosen, mobilen Systemen und die Entwicklung von Lösungen zur Erfüllung der Anforderungen</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse in der Datenkommunikation.
Literatur	<ul style="list-style-type: none"> <li>• Folien zur Vorlesung / Lecture Slides</li> <li>• J. Schiller: Mobile Communications, 2. Auflage, Addison Wesley, 2004</li> <li>• W. Stallings: "Wireless Communications and Networks"; Pearson, 2nd Ed., 2014</li> <li>• Weitere Spezialliteratur wird in den Vorlesungsfolien bekannt gegeben</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

— Communication

+ Mobile Internet Technology (1212346)

Modulverantwortung	Universitätsprofessor Dr.-Ing. Klaus Wehrle & Dr. rer. nat. Dirk Thißen
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Mobile Internet Technology (121234602)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Mobile Internet Technology (121234601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Mobile Internet Technology	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Mobile Radio Networks 1 (Wahlpflichtfach)
Kennung	6010380
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Sommersemester 2011
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• radio wave propagation models</li> <li>• concepts of cellular systems</li> <li>• handover mechanisms</li> <li>• system architecture of modern cellular systems</li> <li>• logical and physical channels</li> <li>• system security in voice networks</li> </ul>
Lernziele/Lernergebnisse	<p>Specific skills:</p> <ul style="list-style-type: none"> <li>• Students will gain knowledge of the possibilities and limits of mobile radio communications and explore them based on current models</li> <li>• Students will learn the basic terminology for relevant topics in the area of telecommunications</li> <li>• Students will understand the design space of cellular networks, especially with respect to radio propagation and systems engineering</li> </ul> <p>General skills:</p> <ul style="list-style-type: none"> <li>• Students will broaden their knowledge of multi-parametric systems design</li> <li>• Students will extend their knowledge of mathematical modelling through abstraction</li> <li>• Students will be able to gauge the basic feasibility of different communication technologies and network designs for specific application areas, e.g. broadband multimedia traffic vs. machine-to-machine low rate traffic</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basic knowledge in the area of communication networks and electromagnetic wave propagation
Literatur	<ul style="list-style-type: none"> <li>• Theodore S. Rappaport, Wireless Communications: Principles and Practice (2nd Edition), Prentice Hall</li> <li>• Y.B. Lin and I. Chlamtac, Wireless and Mobile Network Architectures, Wiley</li> <li>• R. Steele, Mobile Radio Communications, Wiley</li> <li>• J. Eberspächer et al., GSM Switching, Services and Protocols (2nd Edition), Wiley.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	written examination
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Marina Petrova
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	90



— Communication

+ Mobile Radio Networks 1 (6010380)

Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam Mobile Radio Networks 1 (601038001)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture and Exercise Mobile Radio Networks 1	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Mobile Radio Networks 2 (Wahlpflichtfach)
Kennung	6010432
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2011
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Principles of modern cellular systems, e.g. 3G and 4G (LTE) technologies</li> <li>• Fundamentals of modern multi-user wireless transmission techniques, e.g. spread-spectrum, CDMA, OFMDA technologies</li> <li>• Design and capacity estimation of cellular networks</li> <li>• IEEE 802.11 wireless LAN standard</li> <li>• Short-range wireless standards, e.g. Bluetooth</li> </ul>
Lernziele/Lernergebnisse	<p>Specific skills:</p> <ul style="list-style-type: none"> <li>• Students will acquire knowledge of advanced medium access control techniques</li> <li>• Students will learn the basic terminology for relevant topics in the area of modern mobile radio networks and standards</li> <li>• Students will be able to reason about design decisions for enhancing system capacity in modern radio networks</li> </ul> <p>General skills:</p> <ul style="list-style-type: none"> <li>• Students will broaden their knowledge on multi-parametric systems design</li> <li>• Students will extend their knowledge on mathematical modelling through abstraction</li> <li>• Students will be able to assess the basic feasibility of different communication technologies and network designs for specific application areas, e.g. broadband multimedia traffic vs. machine-to-machine low rate traffic</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>• Basic knowledge in the area of communication networks and electromagnetic wave propagation</li> <li>• Basic knowledge in signal and systems</li> </ul>
Literatur	<ul style="list-style-type: none"> <li>• Juha Heiskala and John Terry, OFDM Wireless LANs: A theoretical and Practical Guide, Sams</li> <li>• H. Holma and A. Toskala, WCDMA for UMTS: HSPA Evolution and LTE, Wiley</li> <li>• Matthew Gast, 802.11 Wireless Networks: The Definitive Guide, O'Reilly</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	written examination (90min) or oral examination (30min)
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Petri Mähönen
ECTS Credits	4
Kontaktzeit (SWS)	3

— Communication

+ Mobile Radio Networks 2 (6010432)

Prüfungsdauer (min)	90 oder 30
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam Mobile Radio Networks 2 (601043201)	2. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture and Exercise Mobile Radio Networks 2	2. Semester	keine Semesterempfehlung	-	3

## — Communication

## + Research Focus Class on Communication Systems (1212347)

Modultitel	Research Focus Class on Communication Systems (Wahlpflichtfach)
Kennung	1212347
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2021
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Dieser forschungsorientierte Kurs richtet sich an Studierende, die Interesse an aktuellen Forschungen und Entwicklungen im Bereich der Internet-Technologie haben. Es wird ein ausgewähltes Thema aus einem der folgenden Bereiche der Kommunikationssysteme diskutiert:</p> <ul style="list-style-type: none"> <li>• Netzwerkarchitekturen</li> <li>• Cyber-Physical Systems</li> <li>• Sicherheit und Privatsphäre</li> <li>• Systems Analysis.</li> </ul> <p>Es erfolgt zunächst eine Einführung in den aktuellen Stand der Forschung zum ausgewählten Themenbereich. Im Folgenden soll jeder Studierende unter Anleitung eine Fragestellung (Forschungsidee) im Themenbereich identifizieren und sich genauer in diese einarbeiten und den anderen Teilnehmern präsentieren. Nach dieser Konzeptphase folgt eine Praxisphase, in der die Studierenden ihre Forschungsidee ausarbeiten (prototypische Implementierung, Analyse, Simulation, ...). Der genaue Ablauf kann von Semester zu Semester variieren, je nach Themenbereich.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <ul style="list-style-type: none"> <li>• Detaillierte Kenntnis aktueller Forschungsthemen im Bereich der Kommunikationssysteme und der verteilten Systeme</li> </ul> <p>Fertigkeit:</p> <ul style="list-style-type: none"> <li>• Kenntnis wissenschaftlicher Methoden zur Erarbeitung eigener Ergebnisse, sowohl theoretisch als auch praktisch</li> </ul> <p>Kompetenzen:</p> <ul style="list-style-type: none"> <li>• Vertiefte Kenntnis eines ausgewählten, aktuellen Themas</li> <li>• Fähigkeit zur eigenständigen Einarbeitung in ein Forschungsthema</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Kenntnisse aus der Vorlesung "Advanced Internet Technology" sind hilfreich.
Literatur	Aktuelle Literatur zu ausgewählten Themen wird in der Veranstaltung bekanntgegeben.
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Mündliche Prüfung (20 %); Referat (40 %); Praktikum (40 %). Voraussetzung für die Zulassung zu den Teilprüfungen ist das Bestehen von Hausaufgaben.
Sonstiges	-

— Communication

+ Research Focus Class on Communication Systems (1212347)

Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Research Focus Class on Communication Systems (121234704)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Research Focus Class on Communication Systems (121234701)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Research Focus Class on Communication Systems	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Actions and Planning in AI: Learning, Models, and Algorithms (Wahlpflichtfach)
Kennung	1228568
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Models and Solvers in AI</li> <li>• State models and heuristic search</li> <li>• Logic, SAT solving, and ASP solving</li> <li>• Classical planning: ; language, model, basic algorithms</li> <li>• Markov Decision Processes (MDPs): basic models and algorithms (VI, PI, RTDP, MCTS).</li> <li>• Deep learning (DL) as model and solver; supervised learning, ; Approx VI, PI, MPI with DL.</li> <li>• Reinforcement Learning (RL): Value-based and policy gradient methods</li> <li>• Current research in planning and RL.</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: on completion, the students have ; acquired ; detailed knowledge on</p> <ul style="list-style-type: none"> <li>• models and algorithms for planning,</li> <li>• models and algorithms for reinforcement learning, and</li> <li>• models and algorithms for reinforcement learning that use deep neural networks.</li> </ul> <p>Skills: on completion, the students know</p> <ul style="list-style-type: none"> <li>• how these various algorithms work</li> <li>• when they can be applied, and</li> <li>• how they can be applied.</li> </ul> <p>Competences : on completion, the students are able to</p> <ul style="list-style-type: none"> <li>• model and solve sequential decision problems,</li> <li>• formulate and solve sequential decision problems by trial and error when model is not known</li> <li>• formulate and solve sequential decision problems over large state states using deep nets.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>• basic knowledge of probability theory and logic</li> <li>• basic knowledge of algorithms, graphs, and data structures</li> </ul>
Literatur	<ul style="list-style-type: none"> <li>• S. Russell and P. Norvig. AI: A Modern approach, 4th edition, 2021</li> <li>• R. Sutton and A. Barto. Reinforcement learning: An introduction. 2nd Edition, 2018</li> <li>• H. Geffner, B. Bonet. A Concise Introduction to Models and Methods for Automated Planning. 2013</li> <li>• Various papers.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

– Data and Information Management

+ Actions and Planning in AI: Learning, Models, and Algorithms ...

Modulverantwortung	Prof. Hector Geffner
ECTS Credits	6
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Vorlesung Actions and Planning in AI: Learning, Models, and Algorithms (122856801)	2. Semester	keine Semesterempfehlung	6	0
Übung Vorlesung Actions and Planning in AI: Learning, Models, and Algorithms (122856802)	2. Semester	keine Semesterempfehlung	0	2

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Actions and Planning in AI: Learning, Models, and Algorithms	2. Semester	keine Semesterempfehlung	-	4

Modultitel	Advanced Process Mining (Wahlpflichtfach)
Kennung	1220136
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2019
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Process Mining bietet ein neues Tool zur Verbesserung von Prozessen in einer Vielzahl von Anwendungsbereichen. Es gibt zwei Haupttreiber für diese neue Technologie. Zum einen werden immer mehr Vorfälle aufgezeichnet und liefern so detaillierte Informationen über die Prozesshistorie. Andererseits besteht in den meisten Unternehmen die Notwendigkeit, die Prozessleistung (z.B. zur Reduzierung von Kosten und Durchlaufzeiten) und die Compliance (z.B. zur Vermeidung von Abweichungen oder Risiken) zu verbessern. Process Mining schließt die Lücke zwischen modellbasierten Prozessanalysen (z.B. Simulation, Modellprüfung und klassischen BPM-Techniken) und datenorientierten Techniken (z.B. Data-Mining-Techniken wie Klassifizierung, Clustering und Regression). Process-Mining-Techniken können in einer Vielzahl von Bereichen eingesetzt werden.   Schriftliche Hausarbeit (PM Assignment 1) behandelt praktische Erfahrungen mit Process Mining und die Bewertung von Process-Mining-Techniken und der Wirkung von Rauschen. Schriftliche Hausarbeit (PM Assignment 2) behandelt praktisches Process Mining auf Basis des realen Ereignisprotokolls.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Nach erfolgreichem Abschluss dieses Moduls sollten die Studierenden in der Lage sein: verschiedene Klassen von Petrinetzen zu verstehen und zu begreifen, z.B. (relaxierte) sound Petrinetze, fortgeschrittene Prozesserkennungsalgorithmen tiefgehend zu verstehen, verschiedene Parametrisierungen der Berechnung von Alignments von Prozessmodellen und Ereignisdaten zu verstehen, die Prinzipien des Decomposed Process Mining und die Prinzipien des Stream Based Process Mining zu verstehen.   Fertigkeiten: Die Studenten sollten in der Lage sein, die verschiedenen Arten von Process Mining Tools, die in einer Vielzahl von Process Mining Tools wie ProM, RapidProM, PM4Py, Disco, Celonis, ProcessGold, etc. verfügbar sind, anzuwenden. Darüber hinaus sind die Studierenden in der Lage, die erzielten Ergebnisse und Gründe für die Grenzen der Anwendbarkeit der verschiedenen Werkzeuge und Algorithmen kritisch zu analysieren.   Kompetenzen: Basierend auf den in diesem Kurs erworbenen Kenntnissen und Fähigkeiten sollten die Studenten in der Lage sein, fortgeschrittene Process Mining Algorithmen auf reale industrielle Probleme und Datensätze anzuwenden und verschiedene damit zusammenhängende Fragen von Geschäftsinhabern zu beantworten, die mit dem Prozess zusammenhängen.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>• Basic Knowledge of Process Mining Concepts</li> <li>• Basic Knowledge of Discrete Mathematics</li> <li>• Basic Knowledge of Petri nets</li> </ul>
Literatur	<p>Examples of mandatory papers that need to be studied in detail:  J.M.E.M. van der Werf, B.F. van Dongen, C.A.J. Hurkens, and A. Serebrenik. Process Discovery using Integer Linear Programming. Fundamenta Informaticae, 94: 387-412, 2010.  W.M.P. van der Aalst, A. Adriansyah, and B. van Dongen. Replaying History on Process Models for Conformance Checking and Performance Analysis. WIREs Data Mining and Knowledge Discovery, 2(2):182-192, 2012.  Selected parts of A. Adriansyah, B. van Dongen, and W.M.P. van der Aalst. Memory-Efficient Alignment of Observed and Modeled Behavior. BPM Center Report BPM-13-03, 2013  W.M.P. van der Aalst. Decomposing Petri Nets for Process Mining: A Generic Ap-proach. Distributed and Parallel Databases, 31(4):471-507, 2013.</p>



– Data and Information Management  
+ Advanced Process Mining (1220136)

The textbook "W.M.P. van der Aalst. Process Mining: Data Science in Action. Springer-Verlag, Berlin, 2016" (<http://springer.com/9783662498507>) is advised as background information. Additional background information (optional, just for context or clarification):  
W.M.P. van der Aalst. Process Mining. Communications of the ACM, 55(8):76-83, 2012.  
IEEE Task Force on Process Mining. Process Mining Manifesto. 2011. <http://www.win.tue.nl/ieeetfpm/>.  
A. Rozinat and W.M.P. van der Aalst. Conformance Checking of Processes Based on Monitoring Real Behavior. Information Systems, 33(1):64-95, 2008.  
W.M.P. van der Aalst and C. Stahl. Modeling Business Processes: A Petri Net Oriented Approach. MIT press, Cambridge, MA, 2011.  
R. Lorenz, J. Desel, G. Juhás. Models from Scenarios. T. Petri Nets and Other Models of Concurrency 7: 314-371, 2013.  
A. Adriansyah. Aligning Observed and Modeled Behavior. PhD Thesis Technische Universiteit Eindhoven, 2014.

Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Schriftliche Hausarbeit (40 %); Klausur (60 %). Voraussetzung zum Bestehen des Moduls ist das Bestehen jeder Teilleistung. Es ist nicht möglich, Teilleistungen in ein Folgesemester zu übertragen. ;
Sonstiges	-
Modulverantwortung	Universitätsprofessor Professor h. c. Dr. h. c. Dr. ir. Wil van der Aalst
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	120
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Advanced Process Mining (Übung) (122013602)	2. Semester	keine Semesterempfehlung	0	2
Advanced Process Mining (Klausur) (122013601)	2. Semester	keine Semesterempfehlung	6	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Advanced Process Mining (Vorlesung)	2. Semester	keine Semesterempfehlung	-	3

Modultitel	Business Process Intelligence (Wahlpflichtfach)
Kennung	1216958
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Dieser Kurs beginnt mit einem Überblick über Ansätze und Technologien, die Eventdaten zur Unterstützung des (Re)design von Geschäftsprozessen verwenden. Anschließend konzentriert sich der Kurs auf Process Mining als Brücke zwischen Data Mining und Unternehmensprozessmodellierung. Business Process Intelligence (BPI) und Process Mining ermöglichen es Ingenieuren, betriebliche Prozesse für eine Vielzahl von Organisationen und Systemen (Produktionssysteme, Krankenhäuser, Banken, High-Tech-Systeme, Regierungen, Elektronikgeschäfte, Transportsysteme, Handelssysteme usw.) zu verstehen, zu diagnostizieren, zu verbessern und zu rationalisieren. Der Kurs deckt die drei Haupttypen des Process Mining ab: Process Discovery, Conformance Check und Entrancement. Der Kurs verwendet viele Beispiele anhand von realen Ereignisprotokollen, um die Konzepte und Algorithmen zu veranschaulichen. Nach Abschluss dieses Kurses ist man in der Lage, Process Mining Projekte durchzuführen und verfügt über ein gutes Verständnis des Bereichs Business Process Intelligence (BPI). Darüber hinaus sind die Studierenden in der Lage, Process-Mining-Techniken in allen möglichen Praxisbereichen, einschließlich Praktika und Masterprojekten, direkt anzuwenden.   Schriftliche Hausarbeit (DS Assignment 1) besteht aus einer Analyse eines realen und/oder synthetischen Datensatzes unter Verwendung der im Kurs angebotenen Techniken und Tools. Diese Aufgabe dient dazu, das Verständnis des Materials zu testen. Schriftliche Hausarbeit (DS Assignment) besteht aus einer Analyse komplexerer Datensätze mit verschiedenen datenwissenschaftlichen Techniken. Dazu gehört die Interpretation der Ergebnisse und die kreative Nutzung mehrerer Ansichten der Daten. Die Klausur besteht aus Fragen, um das theoretische Wissen über die erlernten Algorithmen und Techniken zu testen.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Nach erfolgreichem Abschluss dieses Moduls sollten die Studierenden in der Lage sein: Petri-Netze zu verstehen, grundlegende Prozesserkennungsalgorithmen zu verstehen, zu wissen, wie man eine Prozessinstanz oder ein Ereignisprotokoll mit einem Prozessmodell abgleicht, andere Perspektiven zu berücksichtigen, z.B. Performance-Projektion auf ein Petrinetz, und sich mit Konzepten wie verantwortungsvoller Datenwissenschaft und Big Data im Process Mining vertraut zu machen.   Fertigkeiten: Die Studierenden sollten in der Lage sein, verschiedene Process Mining Tools wie ProM und Disco zu nutzen. Darüber hinaus sind sie in der Lage, Ereignisprotokolle zu filtern und zu verstehen, wie sich verschiedene Parameter eines Algorithmus auf das Ergebnis der Analyse auswirken.   Kompetenzen: Basierend auf den in diesem Kurs erworbenen Kenntnissen und Fähigkeiten sollten die Studenten in der Lage sein, grundlegende Process Mining Algorithmen auf reale industrielle Probleme anzuwenden und damit zusammenhängende Fragen von Unternehmern zu beantworten, die mit dem Prozess zusammenhängen.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Prozessmodellierung, Logik, Programmierung und Datenbanken.
Literatur	<p>Das Lehrbuch "W. van der Aalst. Prozess-Mining: Datenwissenschaft in Aktion. Springer-Verlag, Berlin, 2016" <a href="http://springer.com/9783662498507">http://springer.com/9783662498507</a>   <a href="http://springer.com/9783662498507">http://springer.com/9783662498507</a>   <a href="http://springer.com/9783662498507">http://springer.com/9783662498507</a>  ) ist die primäre Informationsquelle und die Vorträge werden mit den Kapiteln des Buches verknüpft. Den Teilnehmern werden Folien, Übungen, Software und Datensätze zur Verfügung gestellt. Das Coursera MOOC on Process Mining   <a href="https://www.coursera.org/learn/process-mining">https://www.coursera.org/learn/process-mining</a>   <a href="https://www.coursera.org/learn/process-mining">https://www.coursera.org/learn/process-mining</a></p>

– Data and Information Management  
+ Business Process Intelligence (1216958)

	<a href="https://www.coursera.org/learn/process-mining">https://www.coursera.org/learn/process-mining</a>   <a href="https://www.coursera.org/learn/process-mining">https://www.coursera.org/learn/process-mining</a>   liefert zusätzliche Hintergrundinformationen, falls die Dinge nicht klar sind.
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Schriftliche Hausarbeit (40 %); Klausur (60 %). Voraussetzung zum Bestehen des Moduls ist das Bestehen jeder Teilleistung. Es ist nicht möglich, Teilleistungen in ein Folgesemester zu übertragen.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Professor h. c. Dr. h. c. Dr. ir. Wil van der Aalst
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Business Process Intelligence (121695802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Business Process Intelligence (121695801)	1. Semester	keine Semesterempfehlung	6	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Business Process Intelligence	1. Semester	keine Semesterempfehlung	-	3

Modultitel	CSCW and Groupware: Concepts and Systems for Computer Supported Cooperative Work (Wahlpflichtfach)
Kennung	1215691
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Mit Groupware oder CSCW (Computer Supported Cooperative Work) befassen sich die Informatik sowie andere Disziplinen mit dem Ziel, die Zusammenarbeit von Arbeitsgruppen zu unterstützen. Aktuelle Entwicklungen führen dazu, daß Groupware, MultiMedia und Telekommunikations- sowie Internetdienste zusammenwachsen. Das Berufsleben künftiger Informatiker/innen wird von der Anwendung und Entwicklung dieser Systeme entscheidend geprägt sein. In diesem Modul werden Konzepte Methoden und Lösungen aus dem Forschungsgebiet CSCW, d.h. der computergestützten Kommunikation und Kooperation vermittelt. Nach einer Einführung in die Grundkonzepte und Methoden sowie die interdisziplinäre Natur des Forschungsgebiets werden folgende Themenschwerpunkte behandelt: kooperatives Dokumentenmanagement und Teamräume, Videokonferenzen und Mediaspace, Unterstützung semi-strukturierte Kooperationsprozesse sowie darauf aufbauend Workflow und Prozessmodellierungen, Architekturen und Algorithmen zur synchronen Bearbeitung gemeinsamer Dokumente und der dabei erforderlichen Integritäts- und Konsistenzsicherung. Ein weiterer Schwerpunkt bildet die Einführung in die Blockchain-Technologie sowie darauf basierender Anwendungen.
Lernziele/Lernergebnisse	Kenntnisse: CSCW Konzepte, Methoden und Theorien; Funktionen und Eigenschaften verschiedener Kooperationsunterstützender Anwendungen; Grundlagen der Blockchain Technologie   Fertigkeiten: Modellierung von Kooperationssystemen; Anforderungsanalyse; Auswahl geeigneter Kooperationslösungen; Bewertung von Blockchain-Technologien   Kompetenzen: Analyse und Bewertung von Kooperationssystemen / - architekturen und deren Anforderungen, Bewertung von Blockchain Lösung und ihrem Einsatz für verschiedene Anwendungsfälle.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	Relevante Papiere werden in einem elektronischen Lernraum zur Verfügung gestellt.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Wolfgang Prinz Ph. D.
ECTS Credits	4
Kontaktzeit (SWS)	3

— Data and Information Management

+ CSCW and Groupware: Concepts and Systems for Computer Supported .

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung CSCW and Groupware (121569102)	1. Semester	keine Semesterempfehlung	0	1
Prüfung CSCW and Groupware (121569101)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung CSCW and Groupware	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Datenstrommanagement und -analyse (Wahlpflichtfach)
Kennung	1226146
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Grundlagen von Datenströmen (Grundkonzepte, Semantik, Abfragen, Fenster, Zeit und Ordnung, Datenmodelle, Abfragesprachen), Komplexe Ereignisverarbeitung</li> <li>• Abfrageverarbeitung (Synopsis, Aggregation, Operator-Implementierung, probabilistische DM) und Optimierung (Operator-Platzierung, algebraische, Subquery-Optimierung)</li> <li>• Datenstromprozessoren und Architekturen (allgemeine Architektur und Komponenten, Kategorisierung, Beispiele: Apache Storm, Azure Data Stream Analytics, Apache Kafka, Apache Flink), Zeitreihen-Datenbanken</li> <li>• Maschinelles Lernen auf Datenströmen (Klassifizierung, Mustererkennung, Ensemble Learning, Anytime Algorithmen, Concept Drift, Dimensionsreduktion)</li> <li>• Metadaten und Datenqualitätsmanagement für Datenströme</li> <li>• Visualisierung von Datenströmen</li> </ul>
Lernziele/Lernergebnisse	<p><b>Kenntnisse</b></p> <ul style="list-style-type: none"> <li>• mehrere Datenmodelle und Abfragesprachen für Datenströme zu nennen</li> <li>• verschiedene Systemtypen und Werkzeuge zur Verarbeitung von Datenströmen aufzulisten</li> <li>• die wichtigsten Aspekte von Zeit und Ordnung zu beschreiben und die Methoden zur Implementierung von Zeit und Ordnung in Datenstromprozessoren zu benennen</li> <li>• Arten von Datenstromoperatoren zu nennen und zu ; beschreiben</li> <li>• Konzepte für Synopsis für Datenströme zu nennen und zu beschreiben</li> <li>• sich ; Methoden zur Datenbereinigung für Datenströme in Erinnerung zu rufen</li> <li>• verschiedene Arten von Algorithmen des maschinellen Lernens für Datenströme zu nennen</li> <li>• die Unterschiede von ML-Algorithmen für Datenströme im Vergleich zu Algorithmen für statische Daten zu nennen</li> <li>• verschiedene Arten von Diagrammen zur Visualisierung von Datenströmen aufzuzählen</li> </ul> <p><b>Fertigkeiten</b></p> <ul style="list-style-type: none"> <li>• verschiedene Arten von Systemtypen und -architekturen zu unterscheiden, zu erklären, wie diese aufgebaut sind, und Vor- und Nachteile zu diskutieren</li> <li>• die Semantik von Datenströmen und entsprechenden Abfragesprachen zu erklären</li> <li>• Operortypen in einer gegebenen Abfrage zu identifizieren und ihre Verwendung zu erklären</li> <li>• mehrere Techniken zur Erstellung von Synopsis zu vergleichen</li> <li>• Datenqualitäts- und Metadatenmanagement-Architekturen für Datenströme zu erklären</li> <li>• einfache Methoden der Datenbereinigung auf Datenströme anzuwenden</li> <li>• Algorithmen des maschinellen Lernens nach ihren Merkmalen zu klassifizieren</li> <li>• allgemeine Methoden zum Trainieren und Testen von ML-Algorithmen auf Datenströmen zu erklären</li> <li>• die Genauigkeit eines gegebenen Analysealgorithmus zu interpretieren</li> <li>• das Prinzip des Concept Drifts zu erklären</li> <li>• Methoden zur Erkennung von Concept Drift in Datenströmen und Anpassung von ML-Algorithmen an Concept Drift zu erläutern</li> <li>• diskutieren, wie die Dimensionalität eines Datenstroms für die Analyse reduziert werden kann</li> <li>• eine optimale Visualisierung für gegebene Datenströme auszuwählen</li> </ul>

## – Data and Information Management

## + Datenstrommanagement und -analyse (1226146)

	<b>Kompetenzen</b> <ul style="list-style-type: none"> <li>• Abfragen für Datenstromprozessoren in den vorgestellten Abfragesprachen zu konstruieren</li> <li>• zu entscheiden, welcher Datenstromprozessor für welche Aufgabe geeignet ist</li> <li>• die Datenqualität von gegebenen Datenströmen zu beurteilen</li> <li>• einfache Methoden der Datenbereinigung in eine Verarbeitungspipeline zu integrieren</li> <li>• den Typ eines ML-Algorithmus für eine gegebene Analyseaufgabe auszuwählen</li> <li>• entscheiden, ob gegebene Datenströme einem Concept Drift unterliegen</li> <li>• eine Verarbeitungspipeline für eine bestimmte Datenstromanalyseaufgabe zu planen und zu entwickeln</li> <li>• die Ergebnisse einer Datenstromanalyseaufgabe zu bewerten und Maßnahmen zur Verbesserung der Ergebnisse vorzuschlagen</li> <li>• Visualisierungen von Datenströmen erstellen und daraus Erkenntnisse ableiten</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Vorläufiger Datenbankkurs, z. B. Datenbanken und Informationssysteme oder Implementierung von Datenbanken. Ein Kurs über maschinelles Lernen oder Datenwissenschaft.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. Dr. Sandra Geisler
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Datenstrommanagement und -analyse (122614601)	2. Semester	keine Semesterempfehlung	6	0

- Data and Information Management
- + Datenstrommanagement und -analyse (1226146)

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung/Übung Datenstrommanagement und - analyse	2. Semester	keine Semesterempfehlung	-	4



Modultitel	Distributed Ledger Technology (Wahlpflichtfach)
Kennung	1226006
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2021
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Diese Lehrveranstaltung hat das Ziel, Grundlagenwissen und praktische Erfahrungen für den Einsatz von Blockchain-Technologien zu vermitteln. Blockchain wird als eine spezifische Instanz der Distributed Ledger Technology (DLT) betrachtet. Die Lehrveranstaltung stellt DLT als neues Paradigma für das Kooperationsmanagement über flexible Geschäftspartnerschaften vor und besteht aus sieben Hauptbausteinen</p> <ul style="list-style-type: none"> <li>• Grundlagen - Verteiltes Transaktionsmanagement versus zentralisierte Datenbanken mit ihren ACID-Eigenschaften für das Konsistenzmanagement werden zuerst diskutiert. Dann wird DLT als Enabler des Internet of Value vorgestellt, als Garant für Ledger zur Vermögensverwaltung sowie für Mikrotransaktionen, die neue Geschäftsfelder eröffnen.</li> <li>• Kerntechnologien - Schlüsseltechnologien zur Etablierung neuer Prozessmuster für die Zusammenarbeit von Unternehmen, um virtuelle Unternehmen und andere Unternehmensinnovationen zu ermöglichen; zu den Schlüsseltechnologien gehören Hashing, Perception Hashes, Merkle-Bäume für die Darstellung von Transaktionen, Konsens für die Prozesssynchronisation sowie Token &amp; Tokenization für die Geschäftsmodellierung.</li> <li>• Use-Case-Portfolio - Eine Auswahl typischer Anwendungsfälle zur Veranschaulichung des Anwendungsspektrums von DLT. Es wird ein Spektrum von Anwendungsfällen vorgestellt, das von der produzierenden Industrie über das Identitätsmanagement bis hin zu selbstverwalteten Organisationsstrukturen und deren Governance-Struktur reicht.</li> <li>• Designparameter - Verschiedene Parameter, z.B. private versus öffentliche Blockchains, charakterisieren Blockchain-Plattformen und die darauf aufbauenden Anwendungen. Darüber hinaus wird ein Entscheidungsrahmen für die Angemessenheit der Blockchain-Technologie für bestimmte Anwendungen vorgestellt.</li> <li>• Blockchain-Implementierungen - Jede Blockchain-Implementierung besteht aus einem Netzwerk von Peers und einer Blockchain-Plattform. Nach einem Überblick über verschiedene Implementierungs-Frameworks, z.B. Ethereum, Hyperledger oder Corda, werden zwei konkrete Plattformen vorgestellt.</li> <li>• Blockchain-Lösungen - Die Blockchain-Technologie ermöglicht es, datenschutzfreundliche Lösungen zu entwerfen, die der Idee der Kundenprofilierung ähneln. Ein herausragendes Beispiel ist das Self-Sovereign-Identity-Management zur Verwaltung und Sicherung von Identitäten mit Verbrauchern auf dem Fahrersitz. Die Verwaltung von Zertifikaten, z. B. für die allgemeine Bildung sowie für den Führerschein, ist ein weiterer prominenter Blockchain-fähiger Dienst.</li> <li>• Prozessautomatisierung und Selbstverwaltung - Smart Contracts sind ein natürliches Mittel für die Automatisierung von Prozessen, sei es für die geschäftliche Zusammenarbeit oder für die Aufrechterhaltung der Konsistenz über Datenräume hinweg. Andererseits ist die Plausibilität und Korrektheit von Programmcode ein wesentlicher Bestandteil. Es werden verschiedene Techniken zur Software-Verifikation diskutiert. Anschließend wird ein Fokus auf die Modellprüfung und die Validierung von in Solidity programmierten Smart Contracts gelegt.</li> </ul>
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <p>Studierende lernen die grundlegenden Technologien einer Blockchain für verteiltes Transaktionsmanagement wie die fälschungssichere Repräsentation von Transaktionen, die</p>

– Data and Information Management  
+ Distributed Ledger Technology (1226006)

	<p>Synchronisation von Knoten im Netz, unterschiedlichen Betreiberformen von öffentlich bis privat als auch Kernanwendungen für die Blockchain-Technologie.</p> <p>;</p> <p>Fertigkeiten:</p> <p>Studierende können für vorgeschlagene Anwendungsszenarien entscheiden, ob sie durch eine Blockchain einen Mehrwert gewinnen oder sie eher klassisch durch eine Datenbank unterstützt werden können. Zudem können sie für eine Blockchain-basierte Anwendung eine Architektur samt technologische Komponenten gestalten. Weiterhin können sie Organisationsformen (Governance) für einen nachhaltigen Betrieb entwerfen.</p> <p>;</p> <p>Kompetenzen:</p> <p>Studierende können nach dieser Veranstaltung methodisch entscheiden, ob eine Idee zur Verbesserung einer Unternehmenskooperation durch eine Blockchain sinnvoll unterstützt werden kann oder ob es eher eine klassische Datenbankanwendung mit Standardbausteinen wie ERP ist. Sie können für eine Blockchain-basierte Lösung Architekturkonzepte samt ihrer technischen Komponenten entwickeln. Zudem sind sie in der Lage eine System Governance für ein nachhaltiges Betriebs- und Betreiberkonzept zu entwickeln.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Die Kenntnisse aus den Lehrveranstaltungen zu eBusiness, CSCW Groupware und Prozessmanagement werden für die Neugestaltung von Prozessen in der Digitalen Wirtschaft hilfreich sein.
Literatur	Die Vorlesungsunterlagen verweisen auf nützliche Literatur.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. Rose & Prof. Prinz
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Distributed Ledger Technology (122600601)	1. Semester	keine Semesterempfehlung	4	0

- Data and Information Management
- + Distributed Ledger Technology (1226006)

Übung Distributed Ledger Technology (122600602)	1. Semester	keine Semesterempfehlung	0	1
---	-------------	--------------------------	---	---

**▲ Angebotsknoten**

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Distributed Ledger Technology	1. Semester	keine Semesterempfehlung	-	2

Modultitel	eBusiness - Anwendungen, Architekturen und Standards (Wahlpflichtfach)
Kennung	1212683
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Die Beherrschung der Digitalen Transformation wird heute als eine Kernherausforderung wirtschaftlicher Änderungsprozesse gesehen, die nicht nur neue Produkte sondern auch neue Verbreitungs- und Geschäftsmodelle hervorbringen (Digitalization). Die Auswirkungen einer Digitalisierung von Produkten und Dienstleistungen sind sehr weitreichend bis hin zu neuen Formen der Betriebs-, Unternehmens- und Regierungsführung (Governance). Wurde der Begriff eBusiness 1998 von IBM ursprünglich als die Transformation von Kernprozessen durch die Nutzung von Internet-Technologien geprägt, so zeichnet er sich heute durch die Schnelligkeit der Änderungsprozesse aus. Die ursprünglich für die Dokumentation von Zahlungstransaktionen entwickelten Block Chains liefern beispielsweise zunächst nur die Technologie für eine verteilte Datenbank. Gleichzeitig ermöglichen Block Chains aber auch tiefgreifende organisatorische Änderungen in Führungs- und Steuerungsaufgaben (Governance), was sie als disruptive Innovation klassifiziert, mit der sich bestehende Geschäftsmodelle ändern werden. So können beispielsweise Melderegister (Ledgers) in ein Peer-to-Peer-Netzwerk übertragen werden ohne den institutionellen Overhead. Die Vorlesung befasst sich zunächst mit den Auswirkungen der Digitalisierung auf Geschäftsprozesse und ihre Geschäftsmodelle. Einführend werden technologische Voraussetzungen (Enabling Technologies) für den elektronischen Datenaustausch, die elektronische online Bezahlung und elektronische Register vorgestellt. Anschließend werden verschiedene Ausprägungen der Digitalisierung gegenübergestellt, die von der Digitalisierung von Informationen bis zu umfassenden soziotechnischen Änderungsprozessen mit ihren Geschäftsmodellen reichen (Change Management). Kundenportale und Plattformen für den Prozessaustausch zwischen Geschäftspartnern werden ebenfalls technologisch diskutiert. Ein abschließender Schwerpunkt liegt auf der Modellierung von Geschäftsmodellen und ihren Entwicklungsprozessen, die heute als Kernelemente jeder Geschäftstransformation gesehen werden.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Nach dieser Vorlesung sind Studierende in der Lage, die Chancen digitaler Technologien und digitaler Prozesse für die Gestaltung neuer Geschäftsmodelle zu erkennen und zu nutzen. Sie kennen Basistechnologien für die Digitalisierung von Geschäftsprozessen im Kontext des elektronischen Datenaustausches, von Bezahlssystemen und verteilter Kassenbücher (Distributed Ledger Technologies). Zudem werden Portale für die direkte Kundenbeziehung als auch B2B-Portale für die Geschäftskooperation vorgestellt. Ein weiterer Schwerpunkt sind Geschäftsmodelle samt ihrer Gestaltung und Modellierung auch um neue Governance-Strukturen zu ermöglichen. Fertigkeiten: Studierende können die Digitalisierung von Prozessen und Daten für die Gestaltung neuer Produktportfolios nutzen, die Auswirkungen digitalisierter Prozesse auf soziotechnische Systeme bewerten, die Integration gemeinsamer Prozesse auf Basis digitalisierter Prozessketten ermöglichen und Geschäftsmodelle für die Digitalisierung neuer Leistungsprozesse bewerten. Kompetenzen: Studierende können digitale Technologien für die Gestaltung neuer oder das Re-Engineering von Prozessen nutzen, können den Mehrwert digitalisierter Prozesse darstellen und den operativen Mehrwert durch Geschäftsmodelle monetarisieren.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse der Informatik.

## – Data and Information Management

## + eBusiness - Anwendungen, Architekturen und Standards (1212683)

Literatur	Folien zur Vorlesung sowie z.B. folgende Bücher:  P. Beynon-Davies (2013). eBusiness, Palgrave Macmillan.  D. Chaffey (2011). E-Business and E-Commerce Management – Strategy, Implementation and Practice (5th edition), Prentice Hall  A. Meier, H. Stormer (2009). eBusiness & eCommerce – Managing the Digital Value Chain, Springer  Kane, G. C., Palmer, D., Phillips, A. N., Kiron, D., & Buckley, N. (2015). Strategy, not technology, drives digital transformation. MIT Sloan Management Review and Deloitte University  Fiet, E. (2014). Conceptualising business models: Definitions, frameworks and classifications. Journal of Business Models, 1(1).  D Chaffey (2014): Digital business and E-Commerce Management , Pearson.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Thomas Rose
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (mündlich/oral)   60-90 (schriftlich/written)
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung eBusiness - Anwendungen, Architekturen und Standards (121268302)	1. Semester	keine Semesterempfehlung	0	1
Prüfung eBusiness - Anwendungen, Architekturen und Standards (121268301)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung eBusiness - Anwendungen, Architekturen und Standards	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Fundamentals of Business Process Management (Wahlpflichtfach)
Kennung	1227457
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2022
Gültig bis	-
Modulniveau	Master
Inhalt	In diesem Modul geben wir eine umfassende Einführung in das Business Process Management (BPM) und vertiefen verschiedene Bereiche des BPM, wie z. B. Prozessmodellierung, Prozessanalyse, Process Mining und Prozess-Redesign.
Lernziele/Lernergebnisse	<p>Kenntnisse:</p> <p>Nach erfolgreichem Abschluss dieses Moduls sollten die Studierenden in der Lage sein, die wichtigsten Strategien und Bereiche des Business Process Managements zu erklären, die in diesem Modul behandelt werden, wie z.B. Prozessmodellierung, Prozessanalyse, Process Mining und Prozess-Redesign.</p> <p>Fertigkeiten:</p> <p>Nach diesem Modul sind die Studierenden in der Lage, Strategien des Business Process Managements auf Organisationen anzuwenden, Geschäftsprozesse in einer formalen Modellierungsnotation zu kommunizieren, einen in einer realen Organisation ablaufenden Prozess zu analysieren, Verbesserungen des Prozesses vorzuschlagen und Prozessdaten zur Untermauerung ihrer Erkenntnisse zu verwenden.</p> <p>Kompetenzen:</p> <p>Auf der Grundlage der in diesem Modul erworbenen Kenntnisse und Fertigkeiten werden die Studierenden in der Lage sein, ihre Ergebnisse in einem unternehmensgerechten Stil zu kommunizieren.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Zu den empfohlenen Vorkenntnissen gehören Logik, Programmierung, Algorithmen und Datenbanken.
Literatur	Fundamentals of Business Process Management, 2nd edition. Die Studierenden werden ermutigt, vor dem Kauf des Buches persönlich zu prüfen, ob eine Notwendigkeit besteht.
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Prüfungsleistungen: schriftliche Übung (20%); schriftliche Übung (30%); schriftliche Prüfung (50%). Die Studierenden müssen alle Prüfungsteile bestanden haben, um das Modul zu bestehen. Es ist nicht möglich, Teile der Prüfungen auf ein anderes Semester zu übertragen.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. ir. Sander Leemans

— Data and Information Management

+ Fundamentals of Business Process Management (1227457)

ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	90-120 min
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Fundamentals of Business Process Management (122745702)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Fundamentals of Business Process Management (122745701)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Fundamentals of Business Process Management	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Implementation of Databases (Wahlpflichtfach)
Kennung	1215692
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Das Modul behandelt die wichtigsten Aspekte der Implementierung von Datenbanksystemen. Dazu gehört die Einführung von Basisarchitekturen (z.B. Schichtenarchitektur) sowie die zur Lösung einzelner Aufgaben notwendigen Verfahren (insbesondere Query-Verarbeitung und Transaktionsmanagement). Die Konzepte der Implementierung werden sowohl auf das klassische relationale Modell als auch auf neuere Datenmodelle (verteilt, objektorientiert, deduktiv, Suchmaschinen) angewendet. Neben dem notwendigen theoretischen Hintergrund werden praktische Konzepte vorgestellt, die es Datenbankadministratoren ermöglichen, Datenbanken effizient zu optimieren.
Lernziele/Lernergebnisse	<b>Kenntnisse:</b> Die Teilnehmer verstehen Datenbankarchitekturen, Algorithmen zur Abfrageverarbeitung und -optimierung, Transaktionsmanagementkonzepte einschließlich Wiederherstellungsalgorithmen und deren Prinzipien sowie die Verwaltung von Datenbanken. <b>Kompetenzen und Fertigkeiten:</b> Die Studierenden sind in der Lage, das Wissen in diesen Bereichen in praktischen Problemen wie dem Aufbau eines Datenmanagementsystems, der Optimierung von Benutzeranfragen, der Auswahl geeigneter Methoden zur Kontrolle und Wiederherstellung der Parallelität anzuwenden. In Teamübungen analysieren und optimieren die Studierenden Datenbankstrukturen und -funktionalitäten und präsentieren ihre eingereichte Lösung vor dem Unterricht. <b>Vorteile für das zukünftige Berufsleben:</b> Fachkenntnisse in der Bewertung, Verwaltung und Optimierung bestehender Datenbanken sowie ein fundiertes Verständnis von Informationssystemarchitekturen in modernen Unternehmen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	D.E. Shasha: Database Tuning - A Principled Approach. Prentice Hall, 1992 Elmasri, S. Navathe: Fundamentals of Database Systems, Addison-Wesley, 4. Aufl. 2003. T. Härder, E. Rahm: Datenbanksysteme – Konzepte und Techniken der Implementierung. Springer 1999. G. Vossen: Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme. Addison-Wesley, 4. Aufl. 2004.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. pol. Matthias Jarke
ECTS Credits	6
Kontaktzeit (SWS)	4



– Data and Information Management  
+ Implementation of Databases (1215692)

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Implementation of Databases (121569202)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Implementation of Databases (121569201)	1. Semester	keine Semesterempfehlung	6	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Implementation of Databases	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Introduction to Bioinformatics (Wahlpflichtfach)
Kennung	1211903
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Genomsequenzen: Algorithmen zum Sequenzvergleich, Sequenzdatenbanken, Muster und Motive, Phylogenetische Bäume. 3D Strukturen und Proteine: 3D Modellierung, Proteindatenbanken, Proteinstrukturanalyse und –prediktion. Proteinexpression und Funktion: DNA Chip Technologie, Genexpressionsanalyse, Clustering, Proteomics. Systeme und Pathways: Metabolische Netzwerke, Pathway-Analyse, Zellsimulation
Lernziele/Lernergebnisse	Kenntnisse: Studenten erlernen die Anwendung der Informatik in den Lebenswissenschaften in den folgenden Bereichen: Alignments und phylogenetische Bäume, Strukturbioinformatik, Proteomics und Systembiologie.   Fertigkeiten: Modellierung natürlicher Phänomene, Verständnis probabilistischer Ansätze.   Kompetenzen: Verständnis der trade-offs und Limitationen algorithmischer Ansätze. vertiefte analytische und logische Fähigkeiten in einem wichtigen Anwendungsfeld.   Die Vorlesung fördert interdisziplinäres Denken: Verständnis informatischer Methoden und Lösungen in der Biologie, insbesondere der Molekularbiologie
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse der Informatik.
Literatur	Reinhard Rauhut, Bioinformatik. Sequenz - Struktur - Funktion. Wiley-VCH, 2001; Richard Durbin, A. Krogh, G. Mitchison, S. Eddy, Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, 1999; Minoru Kanehisa, Post-Genome Informatics. Oxford University Press, 2000.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Thomas Berlage
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (mündlich/oral)   60-90 (schriftlich/written)
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Introduction to Bioinformatics (121190302)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Introduction to Bioinformatics (121190301)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Introduction to Bioinformatics	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Introduction to Data Science (Wahlpflichtfach)
Kennung	1216861
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Ziel des Kurses ist es, einen umfassenden Überblick über die Datenwissenschaft zu geben und die Studierenden mit realen Datensätzen und Tools vertraut zu machen. Der Kurs bietet drei Perspektiven auf die Datenwissenschaft:   Datenwissenschaftliche Infrastruktur, die sich mit Volumen und Geschwindigkeit beschäftigt. Zu den Themen gehören Instrumentierung, große Dateninfrastrukturen und verteilte Systeme, Datenbanken und Datenmanagement sowie Programmierung, und die größte Herausforderung besteht darin, die Dinge skalierbar und sofort zu gestalten.   Data Science Analyse, die sich mit der Extraktion von Wissen aus Daten beschäftigt. Zu den Themen gehören Statistik, Data/Process Mining, Machine Learning Künstliche Intelligenz, Operations Research, Algorithmen und Visualisierung, und die Hauptaufgabe besteht darin, Antworten auf bekannte und unbekannte Unwissenheiten zu geben.   Datenwissenschaftliche Effekte, die sich auf Menschen, Organisationen und die Gesellschaft beziehen. Zu den Themen gehören Ethik &amp; Datenschutz, IT-Recht, Mensch-Technik-Interaktion, Betriebsführung, Geschäftsmodelle, unternehmerisches Handeln, und die größte Herausforderung besteht darin, all dies auf verantwortungsvolle Weise zu tun. Der Kurs wird tiefer in die folgenden Themen eintauchen:   Datenexploration   Datenvisualisierung   Datenqualitätsfragen und -aufbereitung   Datentypen: von Tabellen und Ereignisprotokollen bis hin zu nichtstrukturierten Daten   Beaufsichtigtes Lernen   Entscheidungsbaumpraktiken   Unbeaufsichtigtes Lernen   Clustering   Pattern Mining   Process Mining   Text Mining   Bewertungstechniken   Verteilung mit MapReduce   Verantwortliche Datenwissenschaft: Fairness, Genauigkeit, Vertraulichkeit und Transparenz   Diskriminierungsbewusstes Data Mining   Anonymisierung versus Verschlüsselung Das Ganze wird durch praktische Übungen mit verschiedenen Datensätzen und Softwaretools ergänzt (noch zu bestimmen). Die Schriftliche Hausarbeit (DS Assignment 1) besteht aus einer Analyse eines realen und/oder synthetischen Datensatzes unter Verwendung der im Kurs angebotenen Techniken und Tools. Diese Zuordnung dient dazu, das Verständnis des Materials zu testen. Die Schriftliche Hausarbeit (DS Assignment 2) besteht aus einer Analyse komplexerer Datensätze mit verschiedenen datenwissenschaftlichen Techniken. Dazu gehört die Interpretation der Ergebnisse und die kreative Nutzung mehrerer Ansichten der Daten. Die Klausur besteht aus Fragen, um das theoretische Wissen über die erlernten Algorithmen und Techniken zu testen.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse   Nach dem Kurs sollten die Studierenden einen guten Überblick über das breitere Feld der Datenwissenschaft haben. Durch die praktische Erfahrung mit realen Datensätzen werden die Studierenden die Herausforderungen in den verschiedenen Teilbereichen der Informatik besser verstehen. Die Studenten verstehen visuelle Analytik und fortgeschrittene Ansätze zur Informationsvisualisierung, die Rolle von Big Data und Data Science in der heutigen Gesellschaft, die Grenzen des maschinellen Lernens und Data/Process Mining-Techniken. Darüber hinaus werden einige wenige Themen vertieft und auch theoretische Überlegungen angestellt.   Fähigkeiten   Die Teilnehmer sollten in der Lage sein, kleine Python-Programme zu schreiben und bestehende Programme anzuwenden, Datenvisualisierungs- und Erkundungstechniken durchzuführen, verschiedene Klassifizierungsmethoden aus jedem Datensatz zu konstruieren und die durch überwachtetes Lernen erzielten Ergebnisse auszuwerten, Datenvorverarbeitung durchzuführen und Datenqualitätsprobleme zu erkennen.   Kompetenzen   Basierend auf den in diesem Kurs erworbenen Kenntnissen und Fähigkeiten sollten die Studenten in der Lage sein, die gängigen datenwissenschaftlichen Techniken und die entsprechenden Werkzeuge anzuwenden und die großen Datenherausforderungen und technologischen Ansätze zu kennen.</p>

– Data and Information Management  
+ Introduction to Data Science (1216861)

Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Zu den empfohlenen Vorkenntnissen gehören Logik, Programmierung, Algorithmen und Datenbanken. Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Übungsaufgaben. Details werden in der Vorlesung bekanntgegeben.
Literatur	Jawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", third edition, Morgan Kaufmann Publishers. Some lectures have additional optional literature.
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Schriftliche Hausarbeit (40%); Klausur (60%). Voraussetzung zum Bestehen des Moduls ist das Bestehen jeder Teilleistung. Es ist nicht möglich, Teilleistungen in ein Folgesemester zu übertragen.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Professor h. c. Dr. h. c. Dr. ir. Wil van der Aalst
ECTS Credits	6
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Assessment Introduction to Data Science (121686101)	1. Semester	keine Semesterempfehlung	6	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Introduction to Data Science	1. Semester	keine Semesterempfehlung	-	4
Tutorial Introduction to Data Science	1. Semester	keine Semesterempfehlung	-	2

Modultitel	IT-Sicherheit 2 - Computer Security (Wahlpflichtfach)
Kennung	1211900
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	The lecture comprises of two parts: The first part is the main part of the lecture and is focused on system security. The second part of the lecture is dedicated to privacy in applications. The first part includes: Malware, Buffer Overflows and other Memory Corruptions, Denial of Service Attacks, Access Control, Firewalls, and Intrusion Detection. The second part includes: Anonymous Communications, Electronic Payment, Biometrics, Electronic Voting, and Secure Multi-Party Computation.
Lernziele/Lernergebnisse	Knowledge: On successful completion of this module, students should be able to recall the different types of malware, describe how buffer overflows work and how they can be used by an attacker, recall the different approaches for intrusion detection, recall the different types of firewalls and describe how they work, describe different approaches to protect against buffer overflows, describe the components of a biometric system and the security and privacy implications of different applications of biometric systems, describe how electronic payment methods work, state the challenges of electronic voting systems, describe different electronic voting systems and attacks against them, state the challenges of anonymous communications and explain how different anonymous communication networks work.   Skills: They should be able to identify buffer overflows and other memory corruptions in code and fix them, apply and configure firewalls, apply the Tor protocol, select and compare intrusion detection systems.   Competences: Based on the knowledge and skills acquired they should be able to identify and fix buffer overflows and other memory corruptions, assess the security and privacy of biometric systems as well as electronic payment methods and electronic voting systems. In addition, they should be able to identify weaknesses in protocols that claim to be privacy-preserving.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Cryptographic basics on encryption, integrity protection, signatures and authentication and key agreement. Basic knowledge on operating systems and data communication.
Literatur	Matt Bishop: Introduction to Computer Security, Addison-Wesley.   Ross Anderson: Security Engineering - A Guide to Building Dependable Distributed Systems, Wiley.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessorin Dr.-Ing. Ulrike Meyer
ECTS Credits	6
Kontaktzeit (SWS)	4

– Data and Information Management  
+ IT-Sicherheit 2 - Computer Security (1211900)

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung IT-Security 2 (121190002)	1. Semester	keine Semesterempfehlung	0	1
Prüfung IT-Security 2 (121190001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung IT-Security 2	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Künstliche Intelligenz (Wahlpflichtfach)
Kennung	1215694
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Agent Architecture, Heuristic Search, Games, Knowledge Representation, Bayesian Networks, Machine Learning, Robotics.
Lernziele/Lernergebnisse	Knowledge: Upon successful completion of this module, the student will be familiar with the basic methods underlying the design of intelligent agents, including search methods, knowledge representation using first-order logic, planning, reasoning under uncertainty, and inductive learning.   Skills: The student will be able to apply the methods taught in class to design intelligent agents him- or herself.   Competences: When developing large software systems, the student will be able to identify components and functionalities, which call for the use of Artificial Intelligence methods, and adapt and implement those methods for such purposes.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	Lecture Notes (Transparencies); Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach (2nd Edition), Addison Wesley, 2002.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Gerhard Lakemeyer Ph. D.
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Artificial Intelligence (121569402)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Artificial Intelligence (121569401)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Artificial Intelligence	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Learning Technologies (Wahlpflichtfach)
Kennung	1215751
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Learning Technologies research and engineering is an interdisciplinary field involving competences from cognitive psychology, pedagogy, and various areas within applied computer science. The course consists of two parts: Theoretical foundations about learning and learning technologies, and practical approaches for implementing measures, tools, and techniques related to the theoretical foundations which explore new approaches and develop new learning technologies and methodologies. The theoretical foundations of the course introduces learning theories and their implications to eLearning content and system design. These learning theories are further extended with instructional design theories and multimedia learning theories about motivation and practical principles which are used as foundations for designing eLearning content and describe the eLearning design process. Further topics that are covered in the theoretical foundations include the foundations of assessment and feedback technologies as well as multimedia tools which support the various aspects of the teaching and learning processes in education. The first part of the lecture can be conveyed to the students also by using contemporary e-learning tools and techniques (which include, but not limited to: video based learning, bMOOCS, flipped classroom methodology) The practical part of the lecture takes a look at the current learning technologies and tools present in the current learning processes (but not limited to the learning context), and provides a hands-on approach on building learning technologies tools. The focus is to provide the students a possibility to gain knowledge and build practical skills for innovating and prototyping learning tools, services and systems which can be used in different educational contexts. The practical work and approaches will be connected with new learning theories and approaches which better address the new challenges which come up with the processes for digitalizing the learning environments and the new digital environments within the learning processes.. The course will introduce novel technologies for implementing innovative learning systems and components for open networked learning, and will also include techniques, methodologies and approaches which handle data, data analysis, data literacy and similar novel concepts which arise from using technology within education. The assignments take the form of practical lab courses with mandatory attendance. In the mandatory attendance sessions, the students will do hands-on exercises and receive guidelines, suggestions, and tips about designing and implementing learning tools. As an example, a typical task consists of starting with an idea about tool/component/system, and then iteratively develop it in a finished product, applying agile practical techniques, modern software engineering tools and technologies by following a (user) learner-centered approach. Students can work on these assignments alone, or in small teams.</p>
Lernziele/Lernergebnisse	<p>Upon successful completion of this module, students are able to</p> <p>Knowledge:</p> <ul style="list-style-type: none"> <li>• illustrate the main aspects of current learning theories</li> <li>• describe a systematic process of eLearning content design</li> <li>• explain design principles for multimedia learning by relating them to underlying models and theories of cognitive psychology and pedagogy</li> <li>• give examples of how to apply models of cognitive psychology and instructional design theories in eLearning projects</li> <li>• explain taxonomies of learning objectives by giving appropriate examples</li> <li>• give reasons and examples for eTests and automatic feedback</li> </ul> <p>Skills:</p> <ul style="list-style-type: none"> <li>• Analyze given designs of technology enhanced learning by applying didactic models and multimedia learning design principles</li> </ul>

– Data and Information Management  
+ Learning Technologies (1215751)

	<ul style="list-style-type: none"> <li>• Apply didactic models and multimedia learning design principles when designing and implementing Learning Technologies</li> <li>• Choose and evaluate adequate tools and components for the implementation of technology enhanced learning</li> <li>• Develop ideas and design them as technical solutions in a learning scenario</li> <li>• Evaluate and assess the implementation feasibility of their ideas and designs as learning tools and technologies</li> <li>• Practical agile software development skills which also cover managing software projects work and deliverables</li> <li>• Utilize taxonomies of learning objectives when operationalizing learning objectives and designing test items</li> <li>• Apply principles of assessment and feedback design when implementing test sets</li> <li>• Should have acquired the skills to systematically plan, design and implement small to medium sized eLearning projects</li> </ul> <p>Competences:</p> <ul style="list-style-type: none"> <li>• Based on the knowledge and skills acquired in this module, students will be able to scientifically communicate aspects of eLearning design and eLearning research</li> <li>• Critically discuss learning theories and instructional design theories in the context of requirements for learning technologies development and application</li> <li>• Work in interdisciplinary teams to design and implement technology enhanced learning propose creative solutions in eLearning projects</li> <li>• Take responsibility in project work as a reliable project partner</li> <li>• Identify problems in project work and come up with creative solutions</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Teilnehmer können kleine bis mittelgroße Projekte mit aktuellen Entwicklungsumgebungen umsetzen (i.d.R. Webtechnologien mobile Anwendungen, objektorientierte Programmierung). Sie sind in der Lage, sich schnell in neue Entwicklungsumgebungen und -werkzeuge einzuarbeiten und sind motiviert, die grundlegenden Lerntheorien der Psychologie und Pädagogik und didaktische Modelle zu erlernen.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit (50 %); mündliche Prüfung (50 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen aller schriftlichen Hausaufgaben. In den praktischen Sitzungen der Übung besteht Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Ulrik Schroeder
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Learning Technologies (121575102)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Learning Technologies (121575101)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Learning Technologies	keine Semesterempfehlung	keine Semesterempfehlung	-	3

Modultitel	Privacy Enhancing Technologies for Data Science (Wahlpflichtfach)
Kennung	1212677
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>This module covers current research results in the area of Privacy Enhancing Technologies (PETs) which can be applied to Data Science. These PETs have the potential to enable a new generation of privacy-enabled services which are not focused on maximizing the collection of user data. We use a mix of recent book chapters and papers from conferences and journals of the last few years as primary source material. Based on these sources, the module will cover the following topics:</p> <ul style="list-style-type: none"> <li>• Definitions of privacy and security.</li> <li>• Modelling of privacy threats.</li> <li>• Approaches to anonymization and de-anonymization of data.</li> <li>• Computation on encrypted data.</li> <li>• Approaches for using encrypted data for data science.</li> <li>• Hiding of user queries and data in cloud computing.</li> <li>• Compromises between full encryption and full privacy.</li> <li>• Privacy by design as a cross-cutting software design approach.</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge: After completing this module, students should be able to recall and explain:</p> <ul style="list-style-type: none"> <li>• Privacy threats as defined by standard threat models.</li> <li>• The difference between privacy and security.</li> <li>• Cryptographic techniques which can be used as PETs in the context of data science.</li> </ul> <p>Skills: The students should be able to explain and apply the following techniques:</p> <ul style="list-style-type: none"> <li>• Privacy threat model analysis based on data flow diagrams.</li> <li>• How to apply differential privacy in the context of statistical databases.</li> <li>• Determine which algorithms are suitable for processing encrypted data.</li> <li>• Apply simple baselines for presented approaches.</li> </ul> <p>Competences: Based on the knowledge and skills acquired, the students should be able to:</p> <ul style="list-style-type: none"> <li>• Analyse Data Science approaches embedded in existing systems in regards to privacy threats.</li> <li>• Design new systems using Data Science approaches based on the principles of privacy by design and with the help of the presented PETs.</li> <li>• Assess the advantages and disadvantages of the presented PETs for particular applications of Data Science.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basic knowledge in Data Mining Algorithms and Cryptography are helpful.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

– Data and Information Management

+ Privacy Enhancing Technologies for Data Science (1212677)

Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. pol. Stefan Decker
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	0
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Privacy Enhancing Technologies for Data Science (121267702)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Privacy Enhancing Technologies for Data Science (121267701)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Privacy Enhancing Technologies for Data Science	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Prozess Management (Wahlpflichtfach)
Kennung	1211902
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Die Vorlesung vermittelt Konzepte, Methoden und Sprachen für das Management von Prozessen in Unternehmen und zur Kooperation zwischen Kooperationspartnern. Vorgestellt werden Methoden für die Erfassung, Planung und Ausführung von Prozessen. Präsentiert werden Modellierungsmethoden für verschiedene Anwendungen und Nutzungsszenarien als auch die Ausführungsunterstützung. Zudem werden an Hand von Software-Prozessen Reifegradmodellierung für die Prozessunterstützung diskutiert, um die Qualität von Prozessen bewerten zu können. Die Vorlesung vermittelt: Anforderungen an das Prozess Management, Modellierungskonzepte und Sprachen, Methoden und Werkzeuge für die Ausführungsunterstützung, die Gestaltung und Nutzung von Reifegradmodellen und Individualisierungskonzepte für eine flexible Prozessanpassung.
Lernziele/Lernergebnisse	<b>Kenntnisse:</b> Nach erfolgreicher Durchführung dieses Moduls haben Studierende das Wissen über die Kernkonzepte des Prozessmanagement, verschiedene Sprachen für die Prozessmodellierung mit ihren Zielen und der entsprechenden Modellierungsökonomie, Gestaltungsprinzipien für Optimierungsprojekte, unterschiedliche Reifegradmodelle und das Potential einer formalen Prozessmodellierung zu Leistungsanalyse, Automation und Wiederverwendung   <b>Fertigkeiten:</b> Die Studierenden können unterschiedliche Ziele von Modellierungssprachen differenzieren, die unterschiedlichen Projektdesigns für die Prozessoptimierung differenzieren, die Bedeutung von Reifegradmodellen für die Bewertung und Verbesserung von Prozessqualitäten konzeptualisieren, die verschiedenen Elemente von Prozessdesign bis zur (automatisierten) Ausführung in einer Organisationsumgebung   <b>Kompetenzen:</b> die Studierenden können Modellierungswerkzeuge und Sprachen nutzen für die Formalisierung von Prozessen in verschiedenen Domänen und Kontexten, anwendungsspezifische Beschreibungen in Prozessmodelle transformieren, Prozessspezifikationen mit Workflow-Umgebungen automatisieren, Prozessqualitäten mit Reifegradmodellen hinterlegen und Prozesse zur Optimierung flexibilisieren.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse der Informatik.
Literatur	Folien zur Vorlesung sowie z.B. folgende Bücher: J.v. Brocke, M. Rosemann (2010). Handbook on business process management 1 – Introduction, methods, and information Systems (International Handbooks on Information Systems), Springer  J.v. Brocke, M. Rosemann (2010). Handbook on business process management 2 – Strategic alignment, governance, people and culture (International Handbooks on Information Systems), Springer  M. Weske (2012). Business process management: concepts, languages, architectures, Springer  A. Gadatsch (2012). Geschäftsprozess-Management, Springer Vieweg.  W. van der Aalst, K. van Hee (2004). Workflow management – models, methods and systems, MIT Press  A.-W. Scheer, F. Abolhassan, W. Jost, M. Kirchmer, Eds. (2003) Business process change management, Springer
Sprache	Englisch

– Data and Information Management  
+ Prozess Management (1211902)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Thomas Rose
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (mündlich/oral)   60-90 (schriftlich/written)
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Prozess Management (121190202)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Prozess Management (121190201)	1. Semester	keine Semesterempfehlung	4	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Prozess Management	1. Semester	keine Semesterempfehlung	-	2



Modultitel	Research Focus Class on Learning Technologies (Wahlpflichtfach)
Kennung	1222419
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	Dieser forschungsorientierte Kurs richtet sich an Studierende, die Interesse an aktuellen Fragestellungen, Entwicklungen und Forschung im Bereich moderner Lerntechnologien haben. Es wird ein ausgewähltes Thema aus einem der folgenden Bereiche der Lerntechnologien diskutiert: Learning Analytics, Learning Games, Mobile Learning, Blended Learning, Informatikdidaktik usw. Es erfolgt zunächst eine Einführung in den aktuellen Stand der Forschung zum ausgewählten Themenbereich. Im Folgenden soll jeder Studierende unter Anleitung eine Fragestellung (Forschungsidee) im Themenbereich identifizieren und sich in diese einarbeiten und den anderen Teilnehmern präsentieren. Nach dieser Konzeptphase folgt eine Praxisphase, in der die Studierenden ihre Forschungsidee ausarbeiten (prototypische Implementierung, Analyse, Simulation, usw.) und evaluieren. Der genaue Ablauf kann von Semester zu Semester sowie je nach Themenbereich variieren.
Lernziele/Lernergebnisse	Kenntnisse: Grundlegende Kenntnis aktueller Forschungsthemen im Bereich der Lerntechnologien und/oder Informatikdidaktik. Kenntnis grundlegender Webtechnologien und ihrer Einsatzmöglichkeiten für Lernen und Lehren mittels Technologie.   Fertigkeiten: Kenntnis wissenschaftlicher Methoden zur Erarbeitung eigener Ergebnisse, sowohl theoretisch als auch praktisch.   Kompetenzen: Vertiefte Kenntnis eines ausgewählten, aktuellen Themas; Fähigkeit zur eigenständigen Einarbeitung in ein Forschungsthema.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus den Veranstaltungen "Einführung in Webtechnologien" und/oder "Learning Technologies".
Literatur	Aktuelle Literatur zu ausgewählten Themen; wird in der Veranstaltung bekanntgegeben.
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit oder Praktikum (70 %); Kolloquium (30 %).
Sonstiges	-
Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	2
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	30,0

— Data and Information Management

+ Research Focus Class on Learning Technologies (1222419)

Selbststudium (h)

150,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Research Focus Class on Learning Technologies (122241901)	2. Semester	keine Semesterempfehlung	6	-

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Veranstaltung Research Focus Class on Learning Technologies	2. Semester	keine Semesterempfehlung	-	2

Modultitel	Semantic Web (Wahlpflichtfach)
Kennung	1212675
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Innerhalb der Semantic Web Initiative des W3C wurden Standards für den Austausch von maschinenverständlichen Daten, Informationen und Wissen im Web entwickelt. Diese Standards und Technologien werden nun mehr und mehr in Applikationen und Projekten wie DBPedia, semantic wikis oder kommerziellen Applikationen wie schema.org, der Google's KnowledgeGraph eingesetzt. Diese Vorlesung vermittelt sowohl den theoretischen Hintergrund als auch die praktischen Anwendungen in diesem Gebiet. Unter anderem werden folgende Themen behandelt: Historischer Hintergrund des Semantic Web. RDF Syntax and Datenmodell. RDF Schema und formale Semantik von RDF(S). Ontologien in OWL und formale Semantik von OWL. RDF Datenbanken, Triple-Stores und Wissensbasen und Anfragesprachen. Linked Data Web und Semantic Web Anwendungen.
Lernziele/Lernergebnisse	Kenntnisse: Nach Abschluss des Moduls werden die Studierenden die folgenden Punkte erklären können: Das Wie und Warum von Semantik Web Applikationen   Welche grundlegenden Techniken gibt es und wo kommen sie zur Anwendungen   Wie werden Semantische Daten gespeichert und angefragt?   Welche Wissensrepräsentationsmechanismen gibt es im Semantic Web Bereich? Welche Anwendungen gibt es in welchen Bereichen? Fertigkeiten: Die Studierenden sollten in der Lage sein, die folgenden Techniken zu erklären und anzuwenden: Aufbau von RDF Datenbanken   RDF Anfragen   Ontologien in RDF(S) und OWL. Kompetenzen: Basierend auf dem gewonnenen Wissen und Fähigkeiten sollten die Studierenden in der Lage sein ein System zur Verarbeitung von Semantic Web Daten zu entwerfen und aufzubauen, Datenverarbeitungsabläufe in diesem System einzusetzen und die Vor- und Nachteile von Semantic Web Technologien für Anwendungen abzuwägen.
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Gute Kenntnisse von Datenbanksystemen in Theorie und Praxis.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. pol. Stefan Decker
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	90-120 (schriftlich/written)

Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Semantic Web (121267502)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Semantic Web (121267501)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Semantic Web	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Social and Technological Change (Wahlpflichtfach)
Kennung	1229157
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Der Kurs widmet sich der Klärung grundlegender Fragen wie: Wie können wir die Technologie für uns nutzen? Hat sie in der Vergangenheit für uns funktioniert? Wird sie in Zukunft für uns funktionieren? Gibt es ein monolithisches "Wir"? Was wollen wir eigentlich, kollektiv und individuell? Wie handeln wir, kollektiv und individuell? Wir können diese Fragen nicht beantworten, ohne etwas über Geschichte, Wirtschaft, Politik, Psychologie und Evolutionsbiologie zu wissen. In diesem Kurs werden wir einen solchen Rundgang machen, und teilweise werden wir uns auf Teile der Wahrscheinlichkeits-, Kausalitäts- und Spieltheorie stützen, um unsere Fähigkeiten zum kritischen Denken zu schärfen.</p> <p>Einige der Themen, die in den Vorlesungen behandelt werden, sind:</p> <ul style="list-style-type: none"> <li>• Verständnis der aktuellen Situation: Risiko und Möglichkeiten</li> <li>• Geschichte, Fortschritt, Technologie</li> <li>• Industrielle Revolution und Kapitalismus</li> <li>• Markt und Moral</li> <li>• Der Aufstieg der Vernunft, die Aufklärung</li> <li>• Die Grenzen der Vernunft: Woher kommen die Ziele (Evolution), Dilemmata des kollektiven Handelns (Spieltheorie)</li> <li>• Die Grenzen der Vernunft 2: Kultur, Evolution, soziale Emotionen</li> <li>• KI: Risiken und Chancen</li> </ul>
Lernziele/Lernergebnisse	<p>Wissen: Nach Abschluss des Kurses haben die Studierenden Kenntnisse über den sozialen, wirtschaftlichen, historischen und ethischen Kontext der technologischen Entwicklung in Gegenwart und Vergangenheit erworben.</p> <p>Fähigkeiten: Nach Abschluss des Kurses werden die Studierenden in der Lage sein, über die sozialen Auswirkungen der technologischen Entwicklung nachzudenken, relevante Informationen zu finden und fundiertere Gespräche über dieses Thema zu führen.</p> <p>Kompetenzen: Nach Abschluss des Kurses sind die Studierenden in der Lage, die mit Entscheidungen über die Entwicklung und Regulierung von Technologien verbundenen Probleme zu berücksichtigen.</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Keine.
Literatur	<p>- System error: where big tech went wrong and how we can reboot: Reich, Sahami, Weinstein          - Weapons of math destruction, Cathy O'Neil - Sapiens: Harari - Homo Deus: Harari - Enlightenment Now: S. Pinker - The Mind and the market: Jerry Muller - The tyranny of merit: What's become of the common good? M. J. Sandel - What Money Can't Buy: The Moral Limits of Markets M. J. Sandel - The Technology Trap: Capital, Labor, and Power in the Age of Automation. C B Frey - Alone together, Sherry Turkle - Sustainable development goals (SDGs); UN - Our world in data - Black Mirror (Fernsehserie)</p>
Sprache	Englisch

– Data and Information Management  
+ Social and Technological Change (1229157)

Prüfungsbedingungen	50% schriftliche Hausarbeit, 50% Referat; es besteht Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Prof. Geffner
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Social and Technological Change (122915701)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Social and Technological Change	1. Semester	keine Semesterempfehlung	-	3

Modultitel	The Logic of Knowledge Bases (Wahlpflichtfach)
Kennung	1211393
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	First-Order Logic, The Modal Logic KL, Finite vs. Infinite representability, A Representation Theorem, Only Knowing, Autoepistemic Reasoning, Tractable Reasoning, Situation Calculus
Lernziele/Lernergebnisse	<p>Knowledge: This lecture is about the logical foundations of knowledge bases. At the end of the course the student will be able to characterize the functional view of knowledge bases, distinguish between the knowledge and symbol level, describe why epistemic query languages are needed in the presence of incomplete knowledge, reduce epistemic queries to first-order queries, appreciate the computational complexity inherent in incomplete information.  </p> <p>Skills: The student will be able to use modal logic to analyze the functional and computational requirements of knowledge-based systems, which need to deal with incomplete information.  </p> <p>Competences: The student will be able to play a leading role in the design team of knowledge-based systems.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus Mathematische Logik und/oder Knowledge Representation oder vergleichbare Inhalte.
Literatur	Lecture Notes (Transparencies); Hector J. Levesque and Gerhard Lakemeyer, The Logic of Knowledge Bases, MIT Press, 2001.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Gerhard Lakemeyer Ph. D.
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung The Logic of Knowledge Bases (121139302)	1. Semester	keine Semesterempfehlung	0	2
Prüfung The Logic of Knowledge Bases (121139301)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung The Logic of Knowledge Bases	1. Semester	keine Semesterempfehlung	-	3



Modultitel	Uncertainty in Robotics (Wahlpflichtfach)
Kennung	1222468
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	Covered topics include Introduction to Mobile Robotics, Basics of Probability Theory, State Estimation, Mapping, Markov Localization, Monte Carlo Localization, Simultaneous Localization and Mapping (SLAM), Markov Decision Processes (MDPs), Partially observable Markov Decision Processes (POMDPs), Reasoning about action under uncertainty.
Lernziele/Lernergebnisse	Knowledge: The student will gain an understanding of the sources of uncertainty in mobile robotics. Furthermore, the student will know the basic probabilistic methods underlying state estimation, in particular self-localization, and decision making under uncertainty.   Skills: The student will be able to apply the basic principles of probabilistic state estimation when designing and implementing navigation software components for mobile robots and apply principles of reasoning under uncertainty when designing high-level decision making components for such robots.   Competencies: The student will have the competence to understand and analyze the role uncertainty plays in the various hardware and software components of a mobile robot.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus Einführung in die Künstliche Intelligenz.
Literatur	Sebastian Thrun, Wolfram Burgard, Dieter Fox: Probabilistic Robotics, MIT Press.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %)
Sonstiges	-
Modulverantwortung	Prof. Lakemeyer
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam Uncertainty in Robotics (122246801)	2. Semester	keine Semesterempfehlung	4	0
Exercises Uncertainty in Robotics (122246802)	2. Semester	keine Semesterempfehlung	0	1

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Uncertainty in Robotics	2. Semester	keine Semesterempfehlung	-	2

Modultitel	Wissensrepräsentation (Wahlpflichtfach)
Kennung	1212361
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	First-Order Logic, Resolution, Horn Logic, Procedural Representations, Description Logics, Inheritance Networks, Nonmonotonic Reasoning, Reasoning about Action and Planning
Lernziele/Lernergebnisse	Knowledge: Upon successful completion of this module, the student will be familiar with the basic principles and methods of Knowledge Representation and Reasoning. These include first-order logic and inference by resolution, procedural representations, production systems, description logic, nonmonotonic reasoning, and abduction.   Skills: The student will be able to design knowledge-based systems. In particular, he or she will be able to analyze and cope with the computational complexity of such systems.   Competences: When developing software systems for large applications, the student will be able to identify which parts are best realized using a knowledge-based approach. Moreover, he or she will be able to choose among a number of existing methods to knowledge representation and reasoning and put the chosen methods to practice.
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Kenntnisse aus Mathematische Logik.
Literatur	Lecture Notes (Transparencies; Ron Brachman and Hector J. Levesque, Knowledge Representation and Reasoning, Morgan Kaufmann, 2004.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Gerhard Lakemeyer Ph. D.
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Wissensrepräsentation (121236102)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Wissensrepräsentation (121236101)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Wissensrepräsentation	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Research Focus Class on Data Ecosystems (Wahlpflichtfach)
Kennung	1231482
Version	-
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Dieser forschungsorientierte Kurs richtet sich an Studierende, die Interesse an aktuellen Fragestellungen, Entwicklungen und Forschung im Bereich Datenökosysteme haben. Es wird ein ausgewähltes Thema aus einem der folgenden Bereiche von Datenökosystemen diskutiert: data sovereignty, data exchange, data protection, data security, FAIR data. usw. Es erfolgt zunächst eine Einführung in den aktuellen Stand der Forschung zum ausgewählten Themenbereich. Im Folgenden soll jeder Studierende unter Anleitung eine Fragestellung (Forschungsidee) im Themenbereich identifizieren und sich in diese einarbeiten und den anderen Teilnehmern präsentieren. Nach dieser Konzeptphase folgt eine Praxisphase, in der die Studierenden ihre Forschungsidee ausarbeiten (prototypische Implementierung, Analyse, Simulation, usw.) und evaluieren. Der genaue Ablauf kann von Semester zu Semester sowie je nach Themenbereich variieren.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Nach erfolgreichem Abschluss des Moduls kennen Studierende</p> <ul style="list-style-type: none"> <li>Kenntnisse der wichtigsten Konzepte und aktuellen Trends im Bereich Datenökosysteme. Präsentation des eigenen Forschungsthemas in Form eines wissenschaftlichen Textes, eines Vortrags oder eines Posters. ; Methoden für ein strukturiertes wissenschaftliches Arbeiten an einem gegebenen oder selbstdefinierten Thema. Methoden der systematischen Literaturrecherche und -diskussion zu einem wissenschaftlichen Thema. Konzepte im Forschungsbetrieb (Konferenzen, Journals, andere Veröffentlichungen).</li> </ul> <p>Fähigkeiten: Nach erfolgreichem Abschluss des Moduls können Studierende</p> <ul style="list-style-type: none"> <li>Selbständiges strukturiertes wissenschaftliches Arbeiten (sowohl praktisch als auch theoretisch) an einem gegebenen oder selbstdefinierten Thema. Literaturrecherche und -diskussion zu einem wissenschaftlichen Thema.</li> </ul> <p>Kompetenzen: Auf der Basis der im Modul erworbenen Kenntnisse und Fähigkeiten sind Studierende in der Lage</p> <ul style="list-style-type: none"> <li>Vertiefte Kenntnis eines ausgewählten, aktuellen Themas; Fähigkeit zur eigenständigen Einarbeitung in ein Forschungsthema.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Kenntnisse aus den Veranstaltungen „Datenbanken und Informationssysteme“, „Implementation of Databases“, Data Stream Management and Analysis, oder Seminar „Data Ecosystems“
Literatur	-
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit oder Praktikum (70 %); Kolloquium (30 %).

— Data and Information Management

+ Research Focus Class on Data Ecosystems (1231482)

Sonstiges	-
Modulverantwortung	JProf. Dr. rer. nat. Sandra Geisler
ECTS Credits	6
Kontaktzeit (SWS)	2
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	30,0
Selbststudium (h)	150,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Research Focus Class on Data Ecosystems (123148201)	keine Semesterempfehlung	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Veranstaltung Research Focus Class on Data Ecosystems	keine Semesterempfehlung	keine Semesterempfehlung	-	2

Modultitel	Industrial Data Security (Wahlpflichtfach)
Kennung	1231539
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2025
Gültig bis	-
Modulniveau	Master
Inhalt	<p>This course provides an introduction into current security problems surrounding industrial data and state-of-the-art security solutions. After an introduction into industrial data, how it is stored and processes, as well as prevalent security issues surrounding industrial data, students will learn about approaches to secure industrial data across files, databases, and machine learning. Topics include encryption and integrity-protection of structured industrial data in files, database security in industrial contexts, as well as machine learning security for industrial data. Across all topics, a special emphasis will be put on the privacy-preserving processing of industrial data.</p>
Lernziele/Lernergebnisse	<p>This course will provide students with an understanding of the unique challenges of securely operating on industrial data and provide them with the necessary conceptual tools to properly secure industrial data.</p> <p>Knowledge: Students will be familiar with the security challenges of industrial data, specifically focusing on unique properties of industrial data compared to other settings. Students will know about the various approaches to secure industrial data, covering different aspects of securing industrial data across files, databases, and machine learning.</p> <p>Skills: After taking this course, students will be able to select and apply suitable conceptual tools to realize comprehensive security of industrial data. Based on an assessment of the individual constraints in a given industrial scenario, students will be able to select those security measures that are best applicable and prioritize their deployment.</p> <p>Competences: Students will be able to identify and pinpoint challenges of securing industrial data across different domains. They will further be able to identify where standard security approaches can be adapted to secure industrial data and where dedicated security approaches are required.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Basic knowledge in IT security, databases, and machine learning e.g., as taught in lectures such as "IT-Security", "Databases and Information Systems", and "Elements of Machine Learning and Data Science".
Literatur	Lecture slides. Pointers to further relevant literature will be provided during the lecture.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. Dr. Martin Henze

– Data and Information Management  
+ Industrial Data Security (1231539)

ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Industrial Data Security (123153901)	1. Semester	keine Semesterempfehlung	6	0
Übung Industrial Data Security (123153902)	1. Semester	keine Semesterempfehlung	0	1

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Industrial Data Security	1. Semester	keine Semesterempfehlung	-	3



Modultitel	Linux Kernel Programming (Wahlpflichtfach)
Kennung	1229308
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Advanced C programming techniques</li> <li>• Programming environment for the kernel</li> <li>• Writing a kernel module</li> <li>• Communication in the kernel</li> <li>• Memory management</li> <li>• Virtual File System</li> <li>• System calls</li> <li>• File systems</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge:</p> <ul style="list-style-type: none"> <li>• Advanced C programming techniques (pointer manipulation, error handling, object-oriented programming)</li> <li>• Linux kernel subsystems</li> <li>• Modularity of Linux</li> <li>• Generic Linux kernel data structures</li> <li>• Generic Linux kernel APIs</li> <li>• Memory management in the Linux kernel</li> <li>• Linux Virtual File System (VFS)</li> </ul> <p>Skills:</p> <ul style="list-style-type: none"> <li>• Write kernel modules in C using various Linux APIs</li> <li>• Debug kernel code</li> <li>• Work on a large code base (30M lines of code)</li> <li>• Use various memory management facilities provided by the kernel (memory pools, reference counters, ...)</li> <li>• Implement a user-facing API for your modules (procfs, sysfs)</li> <li>• Use concurrency efficiently in the kernel (work queues, synchronization)</li> </ul> <p>Competencies:</p> <ul style="list-style-type: none"> <li>• Write their own kernel modules (device drivers, file systems, exposing kernel data to user space, ...)</li> <li>• Debug low level systems with debuggers and probes</li> <li>• Work on a large code base and submit patches to mailing lists</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Proficiency in C, 'Operating Systems and System Software' (or equivalent)
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Projekt (50%) und Klausur (50%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.

– Data and Information Management  
+ Linux Kernel Programming (1229308)

Sonstiges	-
Modulverantwortung	Univ.-Prof. Redha Gouicem
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Linux Kernel Programming (122930802)	2. Semester	keine Semesterempfehlung	0	3
Prüfung Linux Kernel Programming (122930801)	2. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Linux Kernel Programming	2. Semester	keine Semesterempfehlung	-	2

Modultitel	Advanced Algorithmic Differentiation (Wahlpflichtfach)
Kennung	1221328
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2021
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• Essential AD</li> <li>• Exploitation of Jacobian and Hessian Sparsity</li> <li>• Limited Memory Adjoint</li> <li>• Elimination Methods for Jacobians</li> <li>• Symbolic Tangents and Adjoint</li> <li>• AD Mission Planning</li> </ul>
Lernziele/Lernergebnisse	<ul style="list-style-type: none"> <li>• Knowledge: understanding of algorithmic differentiation (AD) beyond the black-box perspective</li> <li>• Skills: ability to design scalable and efficient AD solutions</li> <li>• Competences: AD mission planning</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Introduction to Algorithmic Differentiation
Literatur	<ul style="list-style-type: none"> <li>• Set of slides</li> <li>• Example programs</li> <li>• Naumann: The Art of Differentiating Computer Programs. SIAM 2012.</li> <li>• Griewank, Walther: Evaluating Derivatives. SIAM 2008.</li> <li>• References to relevant current literature and online materials</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Uwe Naumann
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Excercise Advanced Algorithmic Differentiation (122132802)	1. Semester	keine Semesterempfehlung	0	2
Exam Advanced Algorithmic Differentiation (122132801)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Advanced Algorithmic Differentiation	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Advanced C++ (Wahlpflichtfach)
Kennung	1228566
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"><li>• development environment<ul style="list-style-type: none"><li>• build system</li><li>• git</li><li>• integrated development environment (IDE)</li><li>• debugging: valgrind, gdb, sanitizer</li><li>• testing: gtest</li><li>• benchmarking: google benchmark</li><li>• compiler explorer (godbolt.org), cppinsights.io, quick-bench.com</li></ul></li><li>• RAII<ul style="list-style-type: none"><li>• object lifetime</li><li>• exception safety</li><li>• smart pointer</li><li>• STL container</li></ul></li><li>• class design<ul style="list-style-type: none"><li>• uniform initialization</li><li>• rule of 5, rule of 0</li><li>• =default, =delete</li><li>• copy elision</li><li>• move semantics</li><li>• rvalue reference</li><li>• universal/forwarding reference</li><li>• std::forward</li></ul></li><li>• specifiers<ul style="list-style-type: none"><li>• auto</li><li>• typedef</li><li>• decltype</li></ul></li></ul>

– Applied Computer Science  
+ Advanced C++ (1228566)

	<ul style="list-style-type: none"> <li>• constexpr, consteval</li> <li>• C++ templates and template type deduction             <ul style="list-style-type: none"> <li>• function templates</li> <li>• class templates</li> <li>• variadic templates</li> <li>• template specialization</li> <li>• SFINAE</li> </ul> </li> <li>• type traits and concepts             <ul style="list-style-type: none"> <li>• type traits</li> <li>• enable_if</li> <li>• remove_cvref</li> <li>• concepts</li> </ul> </li> <li>• expression templates</li> <li>• callables             <ul style="list-style-type: none"> <li>• std::function</li> <li>• lambda</li> <li>• STL iterator</li> <li>• STL algorithm</li> </ul> </li> <li>• parallelization             <ul style="list-style-type: none"> <li>• threads</li> </ul> </li> <li>• error propagation             <ul style="list-style-type: none"> <li>• exceptions</li> </ul> </li> </ul>
Lernziele/Lernergebnisse	<ul style="list-style-type: none"> <li>• Knowledge: understanding of selected advanced concepts and language features provided by modern C++</li> <li>• Skills: ability to write C++ programs that use advanced features of modern C++</li> <li>• Competences: selection of appropriate advanced features of C++ for a given software development problem</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Einführung in die Programmierung mit C++
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Uwe Naumann
ECTS Credits	4

— Applied Computer Science  
+ Advanced C++ (1228566)

Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	60,0
Selbststudium (h)	60,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Advanced C++ (122856601)	2. Semester	keine Semesterempfehlung	4	-

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung + Übung Advanced C++	2. Semester	keine Semesterempfehlung	-	4

Modultitel	Advanced Machine Learning (Wahlpflichtfach)
Kennung	1211912
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Regression techniques, Probabilistic Graphical Models, Exact Inference, Approximate Inference, Deep Generative Models, Deep Reinforcement Learning
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreicher Teilnahme an den Modulveranstaltungen haben die Vorlesungsteilnehmer Kenntnisse und Fähigkeiten in den Themenfeldern, die unter Inhalt beschrieben werden, erworben.   Fertigkeiten: Vorlesungsteilnehmer können Methoden und Techniken, die es einer Maschine ermöglichen, aus Daten zu lernen, herleiten und erklären. Sie kennen die aktuellen Forschungstrends und -entwicklungen. Dadurch sind sie in der Lage, die grundlegenden Machine Learning Techniken, die für diese Fähigkeiten benötigt werden, auszuwählen.   Kompetenzen: Vorlesungsteilnehmer sind in der Lage, die behandelten Methoden selbstständig auf reale Probleme anzuwenden. Sie sind in der Lage, die vorgestellten Algorithmen selbst zu implementieren und diese in einer Programmiersprache ihrer Wahl umzusetzen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse in Linearer Algebra, Wahrscheinlichkeitstheorie und Statistik werden empfohlen. Erfolgreiche Teilnahme an der Vorlesung Machine Learning wird empfohlen.
Literatur	C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.   I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2016.   R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, 2nd Edition, MIT Press, 2018.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. sc. techn. Bastian Leibe
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0



— Applied Computer Science  
+ Advanced Machine Learning (1211912)

Selbststudium (h) 120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Advanced Machine Learning (121191202)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Advanced Machine Learning (121191201)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Advanced Machine Learning	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Automatic Speech Recognition Search (Wahlpflichtfach)
Kennung	1230105
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Large vocabulary speech recognition.</li> <li>• Advanced search methods for large vocabulary continuous speech recognition.</li> <li>• Efficient search space representations.</li> <li>• Current trends in automatic speech recognition.</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>On successful completion of this module, students should be able to:</p> <ul style="list-style-type: none"> <li>• describe the components and formalisms of a state-of-the-art automatic speech recognition system for large vocabulary;</li> <li>• state the optimization problems underlying search for automatic speech recognition for large vocabularies using state-of-the-art automatic speech recognition components and underlying models.</li> </ul> <p>Skills:</p> <p>They should be able to:</p> <ul style="list-style-type: none"> <li>• apply state-of-the-art automatic speech recognition components for large vocabulary;</li> <li>• solve the optimization problems underlying search for automatic speech recognition for large vocabularies using state-of-the-art automatic speech recognition components and underlying models;</li> <li>• should have acquired soft skills like developing and testing software for large vocabulary continuous speech recognition in a cooperative environment.</li> </ul> <p>Competences:</p> <p>Based on the knowledge and skills acquired, they should:</p> <ul style="list-style-type: none"> <li>• have an overview of the state-of-the-art in automatic speech recognition for large vocabulary;</li> <li>• be able to analyze the effect of the components of state-of-the-art automatic speech recognition systems for large vocabulary;</li> <li>• be able to interpret the implementation of an automatic speech recognition system for large vocabulary;</li> <li>• be in a position to realize specific problems of large vocabulary automatic speech recognition.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Das Modul Advanced Methods in Automatic Speech Recognition darf nicht erfolgreich abgeschlossen sein.
(empfohlene) Voraussetzungen	Fundamentals of automatic speech recognition and machine learning.
Literatur	<p>Emphasis on large vocabulary and language modelling:</p> <ul style="list-style-type: none"> <li>• F. Jelinek: Statistical Methods for Speech Recognition. MIT Press, Cambridge, 1997.</li> </ul> <p>Introduction to both speech and language:</p>

## – Applied Computer Science

## + Automatic Speech Recognition Search (1230105)

	<ul style="list-style-type: none"> <li>D. Jurafsky, J. H. Martin: Speech and Language Processing. Prentice Hall, Englewood Cliffs, NJ, 2000.</li> </ul> <p>Advanced topics:</p> <ul style="list-style-type: none"> <li>R. De Mori: Spoken Dialogues with Computers. Academic Press, London, 1998.</li> <li>D. Yu, L. Deng: Automatic Speech Recognition – A Deep Learning Approach. Springer, London, 2015.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Priv.-Doz. Dr. Ralf Schlüter
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Automatic Speech Recognition Search (123010501)	1. Semester	keine Semesterempfehlung	4	0
Übung Automatic Speech Recognition Search (123010502)	1. Semester	keine Semesterempfehlung	0	1

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Automatic Speech Recognition Search	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Combinatorial Problems in Scientific Computing (Wahlpflichtfach)
Kennung	1215721
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Lösung dünnbesetzter linearer Gleichungssysteme</p> <ul style="list-style-type: none"> <li>• Matrix-Vektor Produkte in iterativen Algorithmen</li> <li>• LU Faktorisierung</li> <li>• Cholesky Faktorisierung</li> </ul> <p>Ableitungsberechnung</p> <ul style="list-style-type: none"> <li>• Dynamische Programmierung auf Kettenprodukten erweiterter Jacobimatrizen</li> <li>• Eliminationstechniken auf linearisierten Berechnungsgraphen</li> </ul> <p>Weitere ausgewählte aktuelle kombinatorische Probleme im Wissenschaftlichen Rechnen</p>
Lernziele/Lernergebnisse	<p>The aim of this course is to give students an appreciation of how discrete models are developed in the area of scientific computing. At the end of this course, students should be able to</p> <ul style="list-style-type: none"> <li>• recall the presented combinatorial models in the area of scientific computing (knowledge)</li> <li>• describe the connection between the scientific computing problem and its corresponding combinatorial problem (knowledge)</li> <li>• apply the strategies for the solution of the resulting combinatorial problems (skill)</li> <li>• recognize the expressiveness of different graph models (skill)</li> <li>• discuss combinatorial models for different problem classes arising in scientific computing (competence)</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	<p>Algorithmen und Datenstrukturen</p> <p>Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Übungsaufgaben. Details werden in der Vorlesung bekanntgegeben.</p>
(empfohlene) Voraussetzungen	Algorithmen und Datenstrukturen
Literatur	Vorlesungsskript mit Verweisen auf Originalartikel
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Uwe Naumann</p>
ECTS Credits	4

— Applied Computer Science

+ Combinatorial Problems in Scientific Computing (1215721)

Kontaktzeit (SWS)	3
Prüfungsdauer (min)	0
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Combinatorial Problems in Scientific Computing (121572102)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Combinatorial Problems in Scientific Computing (121572101)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Combinatorial Problems in Scientific Computing	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Computer Vision (Wahlpflichtfach)
Kennung	1215724
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Grundlagen der Bilderzeugung, lineare Filter, Bildsegmentierung, Objekterkennung, Objektkategorisierung, 3D Rekonstruktion, Anwendung aktueller Methoden des maschinellen Lernens für die oben beschriebenen Themen
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreicher Teilnahme an den Modulveranstaltungen haben die Vorlesungsteilnehmer Kenntnisse und Fähigkeiten in den Themenfeldern, die unter Inhalt beschrieben werden, erworben.   Fertigkeiten: Vorlesungsteilnehmer können Methoden und Techniken, die es einer Maschine ermöglichen, Bilder und Videos zu analysieren und ihren Inhalt zu verstehen herleiten und erklären. Sie kennen die aktuellen Forschungstrends und -entwicklungen. Dadurch sind sie in der Lage, die grundlegenden Computer Vision Techniken, die für diese Fähigkeiten benötigt werden, auszuwählen.   Kompetenzen: Vorlesungsteilnehmer sind in der Lage, die behandelten Methoden selbstständig auf reale Probleme anzuwenden. Sie sind in der Lage, die vorgestellten Algorithmen selbst zu implementieren und diese in einer Programmiersprache ihrer Wahl umzusetzen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse in Linearer Algebra, Wahrscheinlichkeitstheorie und Statistik.
Literatur	R. Szeliski, Computer Vision - Algorithms and Applications, Springer, 2010   K. Grauman, B. Leibe, Visual Object Recognition, Morgan & Kaufman publishers, 2011   I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2016   R. Hartley, A. Zisserman. Multiple View Geometry in Computer Vision, 2nd Edition, Cambridge University Press, 2004.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. sc. techn. Bastian Leibe
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0

Selbststudium (h)	120,0
-------------------	-------

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Computer Vision (121572402)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Computer Vision (121572401)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Computer Vision	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Computer Vision 2 (Wahlpflichtfach)
Kennung	1211921
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Single-Object Tracking, Recursive Bayesian Filtering, Multi-Object Tracking, Visual Odometry, SLAM
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreicher Teilnahme an den Modulveranstaltungen haben die Vorlesungsteilnehmer Kenntnisse und Fähigkeiten in den Themenfeldern, die unter Inhalt beschrieben werden, erworben.   Fertigkeiten: Vorlesungsteilnehmer können Methoden und Techniken, die es einer Maschine ermöglichen, Bilder und Videos zu analysieren und ihren Inhalt zu verstehen herleiten und erklären. Sie kennen die aktuellen Forschungstrends und -entwicklungen. Dadurch sind sie in der Lage, die grundlegenden Computer Vision Techniken, die für diese Fähigkeiten benötigt werden, auszuwählen.   Kompetenzen: Vorlesungsteilnehmer sind in der Lage, die behandelten Methoden selbstständig auf reale Probleme anzuwenden. Sie sind in der Lage, die vorgestellten Algorithmen selbst zu implementieren und diese in einer Programmiersprache ihrer Wahl umzusetzen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse in Linearer Algebra, Wahrscheinlichkeitstheorie und Statistik werden empfohlen. Die erfolgreiche Teilnahme an der Vorlesung Computer Vision wird empfohlen.
Literatur	R. Szeliski, Computer Vision - Algorithms and Applications, Springer, 2010.   S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, MIT Press, 2006.   Multiple View Geometry, R. Hartley, A. Zisserman, 2nd edition, Cambridge University Press, 2003   An Invitation to 3D Vision, Y. Ma, S. Soatto, J. Kosecka, S. Sastry, Springer, 2003.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. sc. techn. Bastian Leibe
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Computer Vision 2 (121192102)	1. Semester	keine Semesterempfehlung	0	1
Masterprüfung Computer Vision 2 (121192101)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Computer Vision 2	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Current Topics in Media Computing and HCI (Wahlpflichtfach)
Kennung	1211908
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Dieser Kurs befasst sich mit aktuellen Forschungstrends in den Bereichen Mensch-Computer-Interaktion und Media Computing. Wir verwenden eine Mischung aus aktuellen Buchkapiteln und Beiträgen von Konferenzen und Fachzeitschriften der letzten Jahre, um den Studierenden einen Eindruck von den aktuellen Themen zu vermitteln, an denen in der internationalen Forschungsgemeinschaft gearbeitet wird. Beispiele aus den vergangenen Jahren sind zoombare Benutzeroberflächen, haptische Eingabe- / Ausgabegeräte, Matrixberechnungen für die Modellierung von Benutzeroberflächen und allgegenwärtige Anzeigetechnologien.
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreichem Abschluss dieses Moduls sollen die Studierenden beschreiben können, wie Forschungsprojekte im Bereich Human-Computer Interaction (HCI) etabliert, durchgeführt, evaluiert, veröffentlicht, überprüft und referenziert werden. Darüber hinaus können sie Erkenntnisse und Technologien beschreiben, die aktuelle Trends in der HCI-Forschung darstellen.   Fertigkeiten: Die Studierenden lernen, die Literatur nach Themen im Bereich der HCI zu durchsuchen, Implikationen von HCI-Forschungspublikationen zu lesen und abzuleiten. Darüber hinaus werden sie in der Lage sein, den Wert veröffentlichter HCI-Forschung für die akademische und industrielle Anwendung zu bewerten und zu diskutieren sowie Forschungspublikationen zusammenzufassen und zu überprüfen.   Kompetenzen: Basierend auf den erworbenen Kenntnissen und Fähigkeiten sollten die Studierenden in der Lage sein, mündlich und schriftlich gemäß dem wissenschaftlichen Standard zu kommunizieren.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit (30 %); Klausur oder mündliche Prüfung (70 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Jan Oliver Borchers
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (oral)   90-120 (written)
Gesamtstunden (h)	180,0

— Applied Computer Science

+ Current Topics in Media Computing and HCI (1211908)

Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Current Topics in Media Computing and HCI (121190802)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Current Topics in Media Computing and HCI (121190801)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Current Topics in Media Computing and HCI	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Data Driven Medicine - project-oriented, multidisciplinary introduction (Wahlpflichtfach)
Kennung	1215842
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2021
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Data play an important role in medicine: Intensive care relies on monitors presenting and analysing real-time patient data, medical imaging has become a domain of massive data processing, diagnostics rely on laboratory data, and the importance of data is ever increasing: Wearable sensors, mobile communication devices and respective apps will produce data streams, which support preventive measures in healthy individuals or allow screening as a basis for data-based prevention of diseases. Last but not least: molecular biology (e.g. by gene sequencing and gene expression analysis) introduces new biomarkers, which enable new minimally-invasive diagnostics and approaches to tailoring treatments based on individual characteristics of patients (precision medicine) – which would never be possible without sophisticated processing of huge amounts of data. Medical decision making in general will be markedly influenced by data processing and data analytics. Thus, we can expect data driven medicine to gain momentum in the nearer future. This course offers a project-oriented, multidisciplinary introduction to the basics of data driven medicine. Orientation, fundamental concepts, and methodological approaches are provided by lectures. In addition, the participants will also form small interdisciplinary teams including students of computer science as well as medical students in order to plan and implement an own project, which targets prediction or decision support generated from medical data.</p>
Lernziele/Lernergebnisse	<p>Knowledge: After the course the participants should be able to 1) describe the role of natural language in medical documentation, 2) name and describe relevant methods of predictive analytics and machine learning, 3) name and describe methods for dimensionality reduction, 4) locate data sources for medical big data.   Skills: The participants should be able 1) to extract, load, transform data from relevant medical data sources of medical data via application programming interfaces, 2) access and use medical terminology servers, 3) transform given medical data to standardized representation formats (RDF triplets, i2b2-star-schema), 4) access, design, provide and use medical metadata repositories and ontologies, 5) apply and validate basic machine learning algorithms to medical data   Competences: The participants will acquire the competence to 1) assess data quality of medical data sources, 2) adopt basic methods of natural language processing for providing semantic enrichment and text mining, 3) adopt basic methods of bad data curation, 4) use data to predict important outcomes (predictive modelling).</p>
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	-
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit mit Referat (60 %); Klausur (40 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.

– Applied Computer Science

+ Data Driven Medicine - project-oriented, multidisciplinary ...

Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. pol. Stefan Decker & Univ.-Prof. Dr. med. Rainer Röhrig & Oya Deniz Beyan Ph. D.
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Data Driven Medicine (121584202)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Data Driven Medicine (121584201)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Data Driven Medicine	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Designing Interactive Systems II (Wahlpflichtfach)
Kennung	1215699
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Dieses Modul baut auf den Grundlagen von Designing Interactive Systems I auf und vermittelt ein Verständnis dafür, wie interaktive Multimediasysteme aus Informatiksicht aufgebaut sind. Es behandelt die Prinzipien ereignisbasierter Betriebssysteme, Fenstersystemarchitekturen, Eingabe- und Ausgabegerätetechnologie für mehrere Modalitäten sowie User Interface Management-Systeme und UI-Entwicklungstoolkits und deren Vorzüge zur Gestaltung von UI. In den Übungen werden die Schüler selbst ein minimalistisches Fenstersystem entwickeln, aber auch lernen, mit verschiedenen realen Entwicklungsumgebungen zu arbeiten, einschließlich Fenstersystemen wie Java Swing und Multimedia-Entwicklungsumgebungen, um Benutzeroberflächen zu entwickeln.
Lernziele/Lernergebnisse	Wissen: Nach diesem Kurs werden die Schüler wissen, wie die Technologie hinter interaktiven Systemen funktioniert. Die Studierenden sollen die Architektur klassischer und moderner Fenstersysteme beschreiben können.   Fähigkeiten: Sie können grafische und andere Benutzeroberflächen für vorhandene und neu entstehende Technologien, sowohl für den Desktop als auch darüber hinaus, analysieren, entwerfen und implementieren sowie Schnittstellen für Multimedia-Inhalte einschließen. Gruppenbasierte, projektorientierte Aufgaben und Übungen vermitteln praktische Erfahrungen beim Aufbau von Benutzeroberflächen und fördern das Projektmanagement und die Teamfähigkeit.   Kompetenzen: 50–90% des Entwicklungsaufwands für die heutigen Anwendungen entfallen auf die Benutzeroberfläche. Ein fundiertes Verständnis der Techniken, Vorteile und Fallstricke der verschiedenen in der Industrie verwendeten Ansätze zur Entwicklung von Benutzeroberflächen hilft den Studierenden, fundierte Entscheidungen bei der Implementierung oder Verwaltung von UI-Designprojekten in der Industrie zu treffen, und verschafft ihnen einen entscheidenden Vorsprung bei der Erstellung Neue UI-Architekturen für die schnell wachsenden Märkte von Geräten und Anwendungen der nächsten Generation.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	Eigenes Skript sowie eine Sammlung von Artikeln über grundlegende Systemdesigns.
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit (30 %); Klausur oder mündliche Prüfung (70 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Jan Oliver Borchers
ECTS Credits	6
Kontaktzeit (SWS)	5

— Applied Computer Science  
+ Designing Interactive Systems II (1215699)

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Designing Interactive Systems II (121569902)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Designing Interactive Systems II (121569901)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Designing Interactive Systems II	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Fortgeschrittene Methoden der Virtuellen Realität (Wahlpflichtfach)
Kennung	1212688
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Die Vorlesung behandelt fortgeschrittene Themen der Virtuellen Realität. Basierend auf aktuellen Forschungsarbeiten aus anwendungsorientierten Projekten werden Techniken und Methoden der VR vorgestellt, besprochen und bewertet. Im Einzelnen behandelt die Vorlesung folgende Themen: Methoden der Interaktion und Navigation im dreidimensionalen Raum, multimodale Interaktion einschließlich haptischer und akustischer Interfaces, Immersive Visualisierung als Kombination von wissenschaftlicher Visualisierung und VR, virtuelle Menschmodelle, formale Nutzerstudien
Lernziele/Lernergebnisse	Kenntnisse: Beim erfolgreichen Abschluss des Moduls sollten die Studierenden Kenntnisse über folgende Themen haben: Design, Implementierung und Evaluation von 3D User Interfaces, Aktuelle Entwicklungen in der Interface-Technologie für die Virtuelle Realität, Integration haptischer und akustischer Stimuli in virtuelle Umgebungen, Methoden der Virtuellen Realität In Simulation und Data Science, Generierung und Simulation von virtuellen Menschmodellen sowie Interaktion mit diesen in virtuellen Umgebungen, Systematische Evaluation der Qualität von Benutzerschnittstellen über formale Nutzerstudien   Fertigkeiten: Die Studierenden sollten in der Lage sein: Komplexe, multimodale Interaktionen für immersive VR Systeme zu entwerfen, zu implementieren und zu evaluieren, Aktuelle Forschungsarbeiten der Virtuellen Realität selbstständig nachzuvollziehen, zu re-implementieren, und weiter zu entwickeln   Kompetenzen: Basierend auf dem Wissen und den Fähigkeiten sollten die Studierenden in der Lage sein, VR-Techniken und -Methodik zur Lösung technisch-wissenschaftlicher Problemstellungen zu entwickeln, und auf dem Gebiet der Virtuellen Realität selbstständig wissenschaftlich zu arbeiten
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Basiskenntnisse in Virtueller Realität und Linearer Algebra.
Literatur	D. Bowman et al. 3D User Interfaces. Addison-Wesley   K. M. Stanney. Handbook of Virtual Environments. Erlbaum   M. Slater et al. Computer Graphics & Virtual Environments. Addison-Wesley   G. Burdea, P. Coiffet. Virtual Reality Technology. John Wiley & Sons   K.-F. Kraiss (Ed.). Advanced Man Machine Interfaces. Springer   R.S. Kalawski. The Science of Virtual Reality and Virtual Environments. Addison Wesley
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Torsten Wolfgang Kuhlen
ECTS Credits	6



— Applied Computer Science

+ Fortgeschrittene Methoden der Virtuellen Realität (1212688)

Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Fortgeschrittene Methoden der Virtuellen Realität (121268802)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Fortgeschrittene Methoden der Virtuellen Realität (121268801)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Fortgeschrittene Methoden der Virtuellen Realität	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Geometrieverarbeitung (Wahlpflichtfach)
Kennung	1215696
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Methoden zur Erzeugung von Polygonnetzen (Laserscanning, Registrierung und Integration einzelner Netzteile, etc.), Optimierung von Polygonnetzen (Glättung, Remeshing, Dezimierung, Refinement), Hierarchische Darstellungsformen (coarse-to-fine und fine-to-coarse Hierarchien), Ansätze zur 3D Modellierung mit Netzen, Parametrisierung und Texturierung, Effiziente Datenstrukturen und Netzkompensation.
Lernziele/Lernergebnisse	Bei erfolgreichem Abschluss des Moduls sollten die Studierenden über folgende Kenntnisse verfügen: Techniken zur Erzeugung von hoch-detaillierten dreidimensionalen Modellen von realen Objekten, aktuelle Algorithmen zur Optimierung, Modellierung und Parametrisierung von Geometriedaten mit einem Schwerpunkt auf polygonalen Netzen. Studierende sollen die Fähigkeit erhalten, die erlernten Techniken selbstständig zu implementieren. Basierend auf dem erlernten Wissen und den entwickelten Fähigkeiten sollten Studierende in der Lage sein, Problemstellungen im Bereich der Geometrieverarbeitung zu analysieren, adäquate Algorithmen zur Lösung dieser Probleme auszuwählen und anzuwenden, Stärken und Schwächen der erlernten Algorithmen zu identifizieren sowie Ideen zur Erweiterung und/oder Verbesserung der erlernten Algorithmen zu entwickeln.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Vorlesung „Grundlagen der Computergrafik“, sowie Grundkenntnisse über Algorithmen und Datenstrukturen sowie Lineare Algebra.
Literatur	Botsch, Kobbelt, Pauly, Alliez, Levy: Polygon Mesh Processing
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Leif Kobbelt
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Geometrieverarbeitung (121569602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Geometrieverarbeitung (121569601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Geometrieverarbeitung	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Grundlagen der Computergraphik (Wahlpflichtfach)
Kennung	1212310
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2007
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Grundlagen der Geometriedarstellung (Polygonnetze, Volumendarstellungen, Freiform Kurven und Flächen), Lokale Beleuchtung (3D Transformationen, Clipping, Rasterisierung, Lighting, Shading), Globale Beleuchtung (Sichtbarkeitsproblem, Schattenberechnung, Ray Tracing, Radiosity), Grundlagen der Bildverarbeitung (Transformationen, Farbkodierung, Bildkompression), Volumen-Rendering.
Lernziele/Lernergebnisse	Kenntnis der wichtigsten Datenstrukturen zur Darstellung von dreidimensionalen Objekten und Szenenbeschreibungen. Fertigkeiten: Erlernen der elementaren Operationen und Methoden zur Transformation eines 3D Modells in ein realistisches zweidimensionales Bild (Rendering-Pipeline). Kompetenzen: Überblick über die zentralen Probleme und deren effiziente Lösungen im Bereich der Computer Grafik.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse über Algorithmen und Datenstrukturen sowie Lineare Algebra.
Literatur	Tomas Akenine-Möller et al.: Real-Time Rendering (3rd Edition). Taylor & Francis, 2008   Alan Watt: 3D Computer Graphics (3rd Edition). Addison-Wesley, 1993
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Leif Kobbelt
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Einführung in die Computergraphik (121231002)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Einführung in die Computergraphik (121231001)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Einführung in die Computergraphik	1. Semester	keine Semesterempfehlung	-	3

Modultitel	High-Performance Computing (Wahlpflichtfach)
Kennung	1215720
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• Eigenschaften von Mikroarchitekturen</li> <li>• Parallele Rechnerarchitekturen</li> <li>• Netzwerk-Topologien</li> <li>• Blockalgorithmen zur Ausnutzung von Datenlokalität in tiefen Speicherhierarchien</li> <li>• Prinzipien des parallelen Algorithmenentwurfs</li> <li>• Modellierung von Parallelität (Speedup, Effizienz, Amdahl) und Leistung</li> <li>• Einführung in parallele Programmierung</li> <li>• Weitere ausgewählte Themen</li> </ul>
Lernziele/Lernergebnisse	<p>Erwerb der folgenden Kenntnisse und Fähigkeiten:</p> <ul style="list-style-type: none"> <li>• Verständnis der wesentlichen Parallel-Rechnerarchitekturen</li> <li>• Kenntnis grundlegender Entwurfsmethoden für datenlokale serielle und parallele Algorithmen</li> <li>• Beherrschung einfacher Methoden zur Laufzeitanalyse von parallelen Algorithmen</li> <li>• Grundlegendes Verständnis für elementare Operationen der parallelen Programmierung</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Beherrschung der wesentlichen Konzepte imperativer und objektorientierter Programmiersprachen sowie elementarer Programmiertechniken in diesen Sprachen (Vorlesung Programmierung).
Literatur	<p>PDF-Dateien der Folien und Übungen (zum Download), sowie:</p> <ul style="list-style-type: none"> <li>• G. Hager and G. Wellein: Introduction to High Performance Computing for Scientists and Engineers. CRC Computation Science Series, 2010. ISBN: 978-1-4398-1192-4.</li> <li>• J. Hennessy and D. Patterson: Computer Architecture. A Quantitative Approach. Morgan Kaufmann Publishers, Elsevier, 2011. ISBN: 978-0123838728.</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Matthias Müller</p>
ECTS Credits	6
Kontaktzeit (SWS)	4

— Applied Computer Science  
+ High-Performance Computing (1215720)

Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung High-Performance Computing (121572001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung + Übung High- Performance Computing	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Introduction to Algorithmic Differentiation (Wahlpflichtfach)
Kennung	1221327
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2021
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>We discuss the algorithmic differentiation of differentiable numerical programs, that is the generation of first-, second- and higher-order tangent and adjoint code by</p> <ul style="list-style-type: none"> <li>• operator and function overloading in C++</li> <li>• manual source transformation</li> <li>• automatic source transformation</li> </ul>
Lernziele/Lernergebnisse	<ul style="list-style-type: none"> <li>• Knowledge: understanding of fundamental algorithmic differentiation (AD) modes</li> <li>• Skills: ability to apply AD software to differentiable numerical programs</li> <li>• Competences: choice of appropriate AD mode for a given task</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>• Introduction to Programming</li> <li>• Introduction to Calculus</li> </ul>
Literatur	<ul style="list-style-type: none"> <li>• Set of slides</li> <li>• Example programs</li> <li>• Naumann: The Art of Differentiating Computer Programs. SIAM 2012.</li> <li>• Griewank, Walther: Evaluating Derivatives. SIAM 2008.</li> <li>• References to relevant current literature and online materials</li> </ul>
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Uwe Naumann
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	120
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Introduction to Algorithmic Differentiation (122132702)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Introduction to Algorithmic Differentiation (122132701)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Einführung in Algorithmisches Differenzieren	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Introduction to Numerical Methods and Software with C++ (Wahlpflichtfach)
Kennung	1229154
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	-
Lernziele/Lernergebnisse	<p>Knowledge: understanding of</p> <ul style="list-style-type: none"> <li>- fundamental features of C++</li> <li>- fundamental numerical methods and of their implementation with C++</li> </ul> <p>Skills: ability to implement selected numerical algorithms with C++</p> <p>Competences: expertise in</p> <ul style="list-style-type: none"> <li>- basic C++ programming</li> <li>- algorithmic aspects of numerical methods</li> </ul>
Teilnahmebedingungen (studiengangspezifisch)	-
(empfohlene) Voraussetzungen	<ul style="list-style-type: none"> <li>- Introduction to Calculus</li> <li>- Introduction to Linear Algebra</li> <li>- Introduction to Object-Oriented Programming (e.g. with Java)</li> </ul>
Literatur	Slides, Sample Code, References to current Literature and online materials
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%).
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Uwe Naumann
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

— Applied Computer Science

+ Introduction to Numerical Methods and Software with C++ (1229154)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Introduction to Numerical Methods and Software with C++ (122915401)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung/Übung Introduction to Numerical Methods and Software with C++	1. Semester	keine Semesterempfehlung	-	3

Modultitel	iOS Application Development (Wahlpflichtfach)
Kennung	1215681
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	In diesem Kurs lernen Studierende, wie Sie mobile Anwendungen auf iOS-Geräten entwickeln. Es behandelt folgende Themen: Einführung in die Programmiersprache Swift, Xcode, Storyboards, Model-View-Controller für iOS, App-Frameworks (zB UIKit, Foundation), Debugging mit Instrumenten, Basis-iOS-Entwicklungs-Frameworks (zB MapKit, CoreData, Core Location) , iOS-Grafik- und Spiele-Frameworks (z. B. Sprite Kit, Scene Kit) und Apps im AppStore veröffentlichen.
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreichem Abschluss dieses Moduls sollten die Teilnehmer in der Lage sein, die Struktur eines modernen SDK für mobile Anwendungen zu definieren, die Designrichtlinien für mobile Anwendungen abzurufen und wichtige Konzepte der Softwarearchitektur zu erläutern, die häufig im iOS SDK verwendet werden. Darüber hinaus werden die Unterschiede zwischen mobilem und Desktop-Geräten aufgezeigt und ein Überblick über die vom iOS SDK bereitgestellten Frameworks gegeben.   Fähigkeiten: Studierende werden nach dem Kurs in der Lage sein, ihre eigenen iOS-Apps effektiv zu implementieren, die iOS-Entwicklungsumgebung umfassend zu nutzen und einen iterativen Softwareentwicklungsprozess anzuwenden.   Kompetenzen: Basierend auf den erworbenen Kenntnissen und Fähigkeiten erwerben die Studierenden die Kompetenz, in einem Team zu kommunizieren / zu arbeiten, die Gestaltungsrichtlinien auf ein bestimmtes Anwendungsszenario anzuwenden und einen Entwicklungsplan für eine definierte Anwendung zu erstellen, um ihre Ergebnisse überzeugend zu präsentieren.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundkenntnisse in der objektorientierten Softwareentwicklung.
Literatur	Neuste Version "Programming Fundamentals with Swift" von Matt Neuburg, Verleger: O'Reilly Media
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Referat mit Schriftlicher Hausarbeit (20 %); Projektarbeit mit Referat (50 %); Klausur (30 %). Für die Referate und die Projektarbeit gilt Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Jan Oliver Borchers & Dr. rer. nat. Simon Völker
ECTS Credits	6
Kontaktzeit (SWS)	5

— Applied Computer Science  
+ iOS Application Development (1215681)

Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung iOS Application Development (121568101)	1. Semester	keine Semesterempfehlung	6	0
Übung iOS Application Development (121568102)	1. Semester	keine Semesterempfehlung	0	2

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung iOS Application Development	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Konzepte und Modelle der parallelen und datenzentrischen Programmierung (Wahlpflichtfach)
Kennung	1216838
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>• Architektur von Parallelrechnern (Clustern) zum Einsatz im Hochleistungsrechnen sowie in der Verarbeitung großer Datenmengen (Big Data)</li> <li>• Parallele Programmiermodelle: Instruktionsebene, Beschleuniger, Shared Memory, Distributed Memory, MapReduce-Konzepte</li> <li>• Parallele Verarbeitung von I/O</li> <li>• Synchronisationskonzepte zu den parallelen Programmiermodellen</li> <li>• Realisation häufig verwendeter (abstrakter) Datentypen mit den parallelen Programmiermodellen</li> <li>• Ausgewählte parallele Algorithmen verschiedener Anwendungsbereich</li> <li>• Modellierung von Parallelität (Speedup, Effizienz, Skalierbarkeitsschranken) und Leistung</li> <li>• weitere ausgewählte Themen</li> </ul>
Lernziele/Lernergebnisse	<p>Nach erfolgreicher Teilnahme an den Modulveranstaltungen haben die Studierenden Kenntnisse und Fähigkeiten in den Themenfeldern, die unter Inhalt beschrieben werden, erworben. Somit kennen sie insbesondere die für die parallele Programmierung wesentlichen Eigenschaften von parallelen Rechner-, I/O- und Datenanalysesystemen und deren Architekturen, sowie die Programmiermodelle zur parallelen Programmierung dieser Systeme auf verschiedenen Ebenen. Dies umfasst insbesondere auch die Datenein- und -ausgabe. Für alle diese Ebenen werden die relevanten Konzepte zum Ausdruck von Parallelität und Synchronisation sowie zur Implementierung häufig verwendeter (abstrakter) Datenstrukturen behandelt. Begleitend werden Methoden zum Entwurf und zur Leistungsbewertung der resultierenden Programme vermittelt. Dadurch sind sie in der Lage, parallele Systeme im Einsatz im Hochleistungsrechnen (HPC) sowie zur Verarbeitung großer Datenmengen (Big Data) zu beschreiben und in die Technologieentwicklung einzuordnen. Sie können Optimierungs- und Parallelisierungskonzepte erklären, unterscheiden, und beurteilen. Sie kennen eine Auswahl von parallelen Algorithmen für die genannten Systeme aus verschiedenen Anwendungsbereichen, und sie können neue parallele Algorithmen entwerfen und beurteilen. Sie kennen außerdem die wichtigsten Aspekte der Implementierung der verschiedenen Programmiermodelle. Zu den behandelten Algorithmen kennen sie Komplexitätseigenschaften und Skalierbarkeitsgrenzen.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus 'Programmierung'. Kenntnisse aus 'High Performance Computing' sind hilfreich aber nicht notwendig.
Literatur	PDF-Dateien der Folien und Übungen (zum Download) PDF-Dateien als einzelne Handreichungen zu ausgewählten Themen zur Bearbeitung vorab (zum Download)
Sprache	Englisch
Prüfungsbedingungen	Klausur (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.

— Applied Computer Science

+ Konzepte und Modelle der parallelen und datenzentrischen ...

Sonstiges	-
Modulverantwortung	Dr. rer. nat. Christian Terboven & Universitätsprofessor Dr. rer. nat. Matthias S. Müller
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Konzepte und Modelle der parallelen und datenzentrischen Programmierung (121683802)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Konzepte und Modelle der parallelen und datenzentrischen Programmierung (121683801)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Konzepte und Modelle der parallelen und datenzentrischen Programmierung	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Machine Learning (Wahlpflichtfach)
Kennung	1215744
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Basic concepts: Introduction to Probability Theory, Bayes decision Theory   Probability Density Estimation   Discriminative Methods for Classification: Linear discriminants, Support Vector Machines, AdaBoost   Deep Learning: Multi-Layer Perceptrons, Convolutional Neural Networks, Recurrent Neural Networks
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreicher Teilnahme an den Modulveranstaltungen haben die Vorlesungsteilnehmer Kenntnisse und Fähigkeiten in den Themenfeldern, die unter Inhalt beschrieben werden, erworben.   Fertigkeiten: Vorlesungsteilnehmer können Methoden und Techniken, die es einer Maschine ermöglichen, aus Daten zu lernen, herleiten und erklären. Sie kennen die aktuellen Forschungstrends und -entwicklungen. Dadurch sind sie in der Lage, die grundlegenden Machine Learning Techniken, die für diese Fähigkeiten benötigt werden, auszuwählen.   Kompetenzen: Vorlesungsteilnehmer sind in der Lage, die behandelten Methoden selbstständig auf reale Probleme anzuwenden. Sie sind in der Lage, die vorgestellten Algorithmen selbst zu implementieren und diese in einer Programmiersprache ihrer Wahl umzusetzen.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse in Linearer Algebra, Wahrscheinlichkeitstheorie und Statistik.
Literatur	C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.   I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2016.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. sc. techn. Bastian Leibe
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Machine Learning (121574402)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Machine Learning (121574401)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Machine Learning	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Machine Learning with Graphs: Foundations and Applications (Wahlpflichtfach)
Kennung	1227996
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	Graph-structured data is ubiquitous across application domains ranging from chemo- and bioinformatics to image and social network analysis. To develop successful machine learning algorithms, we need techniques that map the rich information inherent in the graph structure to a vectorial representation in a meaningful way—often called graph embeddings. Roughly, the first two-thirds of this class offers an overview of recent results shedding light on graph embeddings' theoretical properties, i.e., their expressivity and generalization performance, focusing on graph neural networks. In the last third, the course overviews interesting applications of graph embeddings, e.g., the prediction of molecular properties or their use in combinatorial optimization. ;
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>The students will learn about the current state of the art in machine learning on graphs. After completing this course, students will have an in-depth overview of machine learning algorithms used for graph data. On the theoretical side, they will have a thorough understanding of neural and non-neural methods for machine learning on graphs and know about their limitations in terms of expressivity and generalization. On the practical side, they will know about typical use cases of machine learning with graphs.</p> <p>Skills:</p> <p>After completing this course, students will be able to prove limitations of machine learning methods for graphs. Further, they will be able to implement state-of-the-art algorithms.</p> <p>Competences:</p> <p>After completing this course, students will be able to</p> <ul style="list-style-type: none"> <li>• assess the limitations of state-of-the-art machine learning algorithms for graphs,</li> <li>• implement state-of-the-art methods, and</li> <li>• apply theoretical insights of machine learning algorithms for graphs to real-world problems</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Solid Knowledge of Linear Algebra und basic knowledge of graph theory.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). ;Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-

— Applied Computer Science

+ Machine Learning with Graphs: Foundations and Applications ...

Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Machine Learning with Graphs: Foundations and Applications (122799601)	2. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung/Übung Machine Learning with Graphs: Foundations and Applications	2. Semester	keine Semesterempfehlung	-	4

Modultitel	Medical Image Processing (Wahlpflichtfach)
Kennung	9014711
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2012
Gültig bis	-
Modulniveau	Master
Inhalt	<p>The instructors of this lecture are composed of electrical engineering, computer science and medicine to emphasize algorithms as well as clinical applications. They give an overview of medical image formation, enhancement, analysis, visualization, and communication with many examples from medical applications. This includes a brief introduction to medical imaging modalities and acquisition systems. Basic approaches to display multi-dimensional biomedical data are introduced. As a focus, image enhancement techniques, segmentation, texture analysis and their application in diagnostic imaging will be discussed. To complete this overview, storage, retrieval, and communication of medical images are also reviewed.</p> <p>In addition, using ImageJ, a Java-based tool developed at the National Library of Medicine, National Institutes of Health (NIH), USA, the students individually solve problems on the computer applying a variety of algorithms. Each unit of 90 min duration starts with a reading that is followed by an exercise.</p> <p>The topics of the practical exercise include:</p> <ol style="list-style-type: none"> <li>1. Image enhancement</li> <li>2. Segmentation</li> <li>3. Fourier transform</li> <li>4. Visualization</li> <li>5. Automatic processing and measurements</li> <li>6. Extending the ImageJ library</li> </ol>
Lernziele/Lernergebnisse	<p>The students understand basic algorithms and methods of image acquisition, processing and analysis, visualization of 2D and 3D data, and image management in the medical domain, where multiple images of the same kind may differ strongly regarding their visual appearance.</p> <p>The students will learn to transfer their knowledge to novel tasks providing competences and knowledge to</p> <ol style="list-style-type: none"> <li>1. identify major processes involved in formation of medical images</li> <li>2. classify the various medical image processing algorithms</li> <li>3. describe fundamental methods of image enhancement</li> <li>4. appraise efficacy and drawbacks of several techniques of image segmentation</li> <li>5. get familiar with the fundamental concepts of texture and shape analysis</li> <li>6. apply Fourier transform and system theory on image processing</li> <li>7. visualize medical 2D, 3D and 4D data</li> <li>8. explain the basic principles of medical image communication</li> </ol>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	This course is given on a Master's level. Basics in mathematics and computer science are required.
Literatur	Deserno TM (ed): Biomedical Image Processing, Springer-Verlag, Berlin, 2011. ISBN: 978-3-642-15815-5.

— Applied Computer Science  
+ Medical Image Processing (9014711)

Sprache	Englisch
Prüfungsbedingungen	None
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Dr. rer. medic. Marion GrandeModellierungsteamverantwortlicher: Vanessa Ziemons M. A.Modulverantwortlicher: apl. Professor Dr. rer. nat. Thomas Deserno
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Examination Medical Image Processing (901471101)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Medical Image Processing	1. Semester	keine Semesterempfehlung	-	2
Exercise Medical Image Processing	1. Semester	keine Semesterempfehlung	-	2

Modultitel	Mikrocontrollerprogrammierung und Fehlersuche (Wahlpflichtfach)
Kennung	1220524
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2019
Gültig bis	-
Modulniveau	-
Inhalt	<p>The course aims to provide a comprehensive overview of methods for programming of microcontrollers and the debugging, respectively. There will be covered different programming techniques, backgrounds about the software development tool-chains best practices about the programming of microcontrollers and different techniques of de-bugging as deep as performing measurements at the copper layer.</p> <p>The course will dive deeper into the following topics:</p> <ul style="list-style-type: none"> <li>- Practical aspects of the compiler tool-chain</li> <li>- Programming language C</li> <li>- Coding guidelines</li> <li>- Runtime behavior</li> <li>- Debugging at the copper layer</li> <li>- Protective Circuits</li> <li>- First embedded insights to model-based software development and testing</li> </ul> <p>The above will be complemented with hands-on assignments using various examples (still to be determined).</p>
Lernziele/Lernergebnisse	After the course student should have a good overview of programming a microcontroller in an embedded environment in a structured manner. The interaction between the microcontroller as device under development and the embedding environment should be understood in a way that debugging malfunctions is fostered.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	To be decided (several books are being reviewed). Probably a collection of book chapters and articles covering the data science spectrum will be used, complemented by slides.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Dr. Ing. André Stollenwerk
ECTS Credits	6
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-

— Applied Computer Science

+ Mikrocontrollerprogrammierung und Fehlersuche (1220524)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Mikrocontrollerprogrammierung und Fehlersuche (Prüfung) (122052401)	2. Semester	keine Semesterempfehlung	6	0
Mikrocontrollerprogrammierung und Fehlersuche (Übung) (122052402)	2. Semester	keine Semesterempfehlung	0	2

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Mikrocontrollerprogrammierung und Fehlersuche (Vorlesung)	2. Semester	keine Semesterempfehlung	-	4

Modultitel	Physikalisch-Basierte Animation (Wahlpflichtfach)
Kennung	1215862
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Grundlagen der physikalisch-basierten Animation, Partikelsysteme, Starrkörper, Simulation deformierbarer Festkörper mit diskreten und kontinuierlichen Modellen, Simulation von Fluiden, Kollisionserkennung und -behandlung
Lernziele/Lernergebnisse	Kenntnisse: Bei erfolgreichem Abschluss des Moduls sollten die Studierenden über folgende Kenntnisse verfügen: Verständnis aktueller Simulationsverfahren für Starrkörper, deformierbare Körper und Fluide, Erfahrung mit Echtzeitsimulation in der Computergraphik, Algorithmen zur Kollisionserkennung   Fertigkeiten: Studierende sollten in der Lage sein die erlernten Techniken selbstständig zu implementieren   Kompetenzen: Basierend auf dem erlernten Wissen und den entwickelten Fähigkeiten sollten Studierende in der Lage sein, Problemstellungen im Bereich der physikalisch-basierten Animation zu analysieren, passende Verfahren zur Lösung eines Problems auszuwählen und anzuwenden, Simulationsverfahren zu bewerten, die erlernten Verfahren durch eigene Ideen zu erweitern
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlegende Kenntnisse von Numerik, Algorithmen und Datenstrukturen, Computergraphik.
Literatur	Wird in der Vorlesung bekannt gegeben.
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Jan Stephen Bender
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Physikalisch-Basierte Animation (121586202)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Physikalisch-Basierte Animation (121586201)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Physikalisch-Basierte Animation	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Physikalische Simulation im Visual Computing (Wahlpflichtfach)
Kennung	1212692
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2022
Gültig bis	-
Modulniveau	Master
Inhalt	Mehrkörpersysteme, Deformierbare Festkörper, Finite-Elemente-Methoden, Volumenerhaltung, Simulation inkompressibler Fluide mit Lagrangeschen Ansätzen, Viskosität, Oberflächenspannung
Lernziele/Lernergebnisse	Kenntnisse: Bei erfolgreichem Abschluss des Moduls sollten die Studierenden über folgende Kenntnisse verfügen: Verständnis fortgeschrittener Techniken für die Simulation von Mehrkörpersystemen, deformierbarer Festkörper und Fluide, Erfahrung mit der Simulation komplexer physikalischer Verhaltensweisen unterschiedlicher Materialien, Verständnis von Methoden zur Kollisionsbehandlung   Fertigkeiten: Studierende sollten in der Lage sein, die erlernten Techniken selbstständig zu implementieren   Kompetenzen: Basierend auf dem erlernten Wissen und den entwickelten Fähigkeiten sollten Studierende in der Lage sein, Problemstellungen im Bereich der physikalisch-basierten Simulation zu analysieren, passende Verfahren zur Lösung eines Problems auszuwählen und anzuwenden, Simulationsverfahren zu bewerten, die erlernten Verfahren durch eigene Ideen zu erweitern
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Keine.
Literatur	Wird in der Vorlesung bekannt gegeben.
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

— Applied Computer Science

+ Physikalische Simulation im Visual Computing (1212692)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Physikalische Simulation im Visual Computing (121269202)	2. Semester	keine Semesterempfehlung	0	1
Prüfung Physikalische Simulation im Visual Computing (121269201)	2. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Physikalische Simulation im Visual Computing	2. Semester	keine Semesterempfehlung	-	3

Modultitel	Reinforcement Learning and Learning-based Control (Wahlpflichtfach)
Kennung	4026526
Version	-
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2022
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Class outline:</p> <ul style="list-style-type: none"> <li>• Reinforcement learning problem and its relation to control</li> <li>• Markov decision process</li> <li>• Dynamic programming</li> <li>• Tabular reinforcement learning</li> <li>• Reinforcement learning with function approximation incl. deep RL</li> <li>• Policy gradient methods</li> <li>• Model learning</li> <li>• Controller learning</li> </ul>
Lernziele/Lernergebnisse	<p>Wissen und Verstehen:</p> <p>The course “Reinforcement Learning and Learning-based Control” covers fundamentals and state-of-the art methods in reinforcement learning and learning-based control. Reinforcement learning (RL) is a machine learning paradigm that aims at learning action or control policies from data generated through interaction with an environment. It is one important subfield of learning-based control (LBC), which more broadly denotes the intersection of the areas of automatic control and machine learning. Both, RL and LBC are very active and interdisciplinary areas of research.</p> <p>The first part of the module introduces and formalizes the reinforcement learning problem and its connections with dynamical systems and control. Building on the formulation as a Markov decision process, core concepts of RL and optimal control will be developed. After establishing understanding and fundamental concepts of RL theory, modern algorithms including deep RL approaches are introduced. In addition to RL, the course covers further essential topics of learning-based control, including model learning and modern controller tuning techniques</p> <p>Fertigkeiten und Kompetenzen:</p> <p>The course combines profound understanding of theoretical foundations with the application of state-of-the-art techniques to engineering problems. Students acquire a solid foundation of reinforcement learning theory and learningbased control. They will gain insight into how machine learning and control techniques can be combined, and what special challenges exist. Further, students will be exposed to state-of-the-art algorithms and methods such as deep reinforcement learning and model learning, including hands-on programming exercises.</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	<p>Empfohlene Voraussetzungen:</p> <p>Basic knowledge in probability theory and supervised machine learning (such as covered in Computer Science in Mechanical Engineering 2 or Machine Learning, for example)</p>
Literatur	Reinforcement Learning: An Introduction, 2nd edition; Richard S. Sutton, Andrew G. Barto; MIT Press; 2018

– Applied Computer Science

+ Reinforcement Learning and Learning-based Control (4026526)

	Empfohlene weiterführende Literatur: Will be announced in the lecture.
Sprache	Englisch
Prüfungsbedingungen	100% grade of the exam.
Sonstiges	-
Modulverantwortung	Univ.-Prof. Dr. Sebastian Trimpe
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Reinforcement Learning and Learning-based Control (402652601)	2. Semester	keine Semesterempfehlung	6	-

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Reinforcement Learning and Learning-based Control	2. Semester	keine Semesterempfehlung	-	2
Übung Reinforcement Learning and Learning-based Control	2. Semester	keine Semesterempfehlung	-	2

Modultitel	Shape Analysis and 3D Deep Learning (Wahlpflichtfach)
Kennung	1223639
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	Shape Representations (meshes, voxel grids, point clouds, implicit functions) Analysing single shapes (feature descriptors, global/partial symmetries, segmentations, decompositions) Maps between two shapes (functional maps) Geometric Deep Learning (suitable representations, autoencoding tasks, learning high-level structures)
Lernziele/Lernergebnisse	<ul style="list-style-type: none"> <li>- Knowledge of different techniques for 3D shape representation and description</li> <li>- Understanding of geometry related aspects and challenges in the domain of deep learning</li> <li>- Capability to implement neural networks for basic geometry related tasks</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	The lecture "Basic Techniques in Computer Graphics" is recommended but not a hard requirement. The lecture "Geometry Processing" is considered helpful, but also not required.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit mit Präsentation (40 %); Klausur oder mündliche Prüfung (60 %). Voraussetzung für die Zulassung zur Klausur/ mündlichen Prüfung ist das Bestehen der Projektarbeit. Um das Modul zu bestehen müssen beide Teilleistungen bestanden werden.
Sonstiges	-
Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam Shape Analysis and 3D Deep Learning (122363901)	2. Semester	keine Semesterempfehlung	6	-
Excercise Shape Analysis and 3D Deep Learning (122363902)	2. Semester	keine Semesterempfehlung	0	2

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture Shape Analysis and 3D Deep Learning	2. Semester	keine Semesterempfehlung	-	3

Modultitel	Simulation Software Engineering (Wahlpflichtfach)
Kennung	8015429
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2009
Gültig bis	-
Modulniveau	Master
Inhalt	Diese Vorlesung behandelt Softwareengineering-Techniken und -Werkzeuge für die Entwicklung paralleler Simulationscodes. - Softwareentwicklungsprozess und -lebenszyklus - Anforderungsanalyse - Entwurf - Qualitätssicherung - Einsatz und Pflege - Wissenschaftliche Bibliotheken
Lernziele/Lernergebnisse	Die Studierenden sollen in die Lage versetzt werden, korrekte, robuste, effiziente und wartbare parallele numerische Software zu schreiben.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Programmiersprache C/C++.
Literatur	I. Sommerville: Software Engineering, Addison-Wesley, 8th edition, 2006. S. Oliveira, D. Stewart: Writing Scientific Software, Cambridge University Press, 2006
Sprache	Englisch
Prüfungsbedingungen	Die Benotung ergibt sich aus 3 Teilen: • Zwei Referate welche jeweils mit 30% gewichtet werden; • Eine schriftliche Hausarbeit welche mit 40% gewichtet wird.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Felix Gerd Eugen Wolf
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0



## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Masterprüfung Simulation Software Engineering (801542901)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Simulation Software Engineering	1. Semester	keine Semesterempfehlung	-	2
Vorlesung Simulation Software Engineering	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Statistische Methoden zur Verarbeitung natürlicher Sprache (Wahlpflichtfach)
Kennung	1215695
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Introduction/Motivation   Linguistic and Statistical Foundations   Text and Document Classification   Language Modelling   Part-of-Speech (POS) Tagging   Information Extraction by Tagging   Probabilistic Context Free Grammars and Parsing   Machine Translation
Lernziele/Lernergebnisse	Knowledge: On successful completion of this module, students should be able to: describe the various applications of advanced state-of-the-art methods of Natural Language Processing.   describe the fundamental properties and methods of Natural Language Processing.   describe the advanced methods for training a Natural Language Processing: system.   describe the trade-off between system complexity and performance in an advanced Natural Language Processing system. Skills: They should be able to: to train the parameters of a Natural Language Processing system using advanced training methods.   apply and implement advanced methods of Natural Language Processing   measure and analyse the performance of a Natural Language Processing   system in complex real-life applications. Competences: Based on the knowledge and skills acquired they should: have an overview of advanced methods in Natural Language Processing.   be able to apply advanced methods of Natural Language Processing.   be in a position to analyze specific problems in a real-life application of Natural Language Processing systems.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus „Einführung in die Stochastik“, „Datenstrukturen und Algorithmen“, „Formale Systeme, Automaten, Prozesse“.
Literatur	C. D. Manning, H. Schütze: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA, 1999.   D. Jurafsky, J. H. Martin: Speech and Language Processing. Prentice Hall, Englewood Cliffs, NJ, 2000.   Folien/Lecture Notes: <a href="http://www-i6.informatik.rwth-aachen.de/web/Teaching/">http://www-i6.informatik.rwth-aachen.de/web/Teaching/</a>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Hermann Ney
ECTS Credits	8
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	240,0

— Applied Computer Science

+ Statistische Methoden zur Verarbeitung natürlicher Sprache ...

Präsenzstunden (h)	90,0
Selbststudium (h)	150,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Statistische Methoden zur Verarbeitung natürlicher Sprache (121569502)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Statistische Methoden zur Verarbeitung natürlicher Sprache (121569501)	1. Semester	keine Semesterempfehlung	8	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Statistische Methoden zur Verarbeitung natürlicher Sprache	1. Semester	keine Semesterempfehlung	-	4

Modultitel	Stochastic Processes and Dynamical Systems (Wahlpflichtfach)
Kennung	4029089
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<p>A wide range of real-world phenomena are subject to dynamical movement, change, and dependency on their past. This behavior is particularly prominent in engineering and computer systems. In particular, when additional stochastic effects are influencing the dynamics, it is necessary to generalize standard concepts from probability theory through an explicit time component. This class aims at conveying a measure theoretical introduction to these advanced topics and we start by introducing conditional expected values with respect to sigma algebras. This is a very general formulation and critical for considering processes that are defined in continuous time and general state spaces. We will consider Markov processes that go beyond the often considered discrete state-space and discrete time Markov chains. Further, we investigate selected special cases such as the famous Wiener process and some of its properties. We will generalize to the Ornstein-Uhlenbeck process and touch upon stochastic differential equations. Stochastic processes and dynamical systems are key concepts in theory and practice. This class covers some of the mathematical foundations for further research-oriented topics that consider a data-driven approach to treating stochastic processes and dynamical systems.</p>
Lernziele/Lernergebnisse	<p>Knowledge and Understanding:</p> <p>This class is mathematically oriented and we consider stochastic processes and dynamical systems from a measure theoretical point of view. Both are fundamental for modeling various modern systems in engineering and computer science. In essence, they describe the evolution of systems that depend on their own past. The mathematical analysis of these systems allows to predict, analyze and control them. Depending on the application, there can be stochastic or deterministic behavior.</p> <p>Skills and Competencies:</p> <p>The key objective is to convey the key ideas and concepts, while also proving and discussing selected results in more detail. The obtained insights are tailored to improve the understanding of stochastic processes from a measure theoretical perspective. Here, we will also focus on standard results and insights. We will cover the special case of linear and time-invariant systems in great detail due to the analytical tractability. The deterministic case will receive special attention for its relevance in modern control theory. Among others, we will cover conditional expected values and Markov processes in various settings (discrete time, continuous time, finite state spaces), where we discuss selected examples such as the Brownian Motion and Ornstein-Uhlenbeck process alongside their properties.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	<p>Recommended Requirements:</p> <p>Basic familiarity with probability theory and machine learning (as taught, e.g., in the class „Foundations of machine learning“ by Prof. Dr. Sebastian Trimpe)</p>
Literatur	-

— Applied Computer Science

+ Stochastic Processes and Dynamical Systems (4029089)

Sprache	Englisch
Prüfungsbedingungen	Mündliche Prüfung, Grippenprüfung
Sonstiges	-
Modulverantwortung	Univ.-Prof. Dr. Johann Sebastian Trimpe
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Oral exam Stochastic Processes and Dynamical Systems (402908901)	1. Semester	2. Semester	4	-

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture with integrated exercises Stochastic Processes and Dynamical Systems	1. Semester	2. Semester	-	3

Modultitel	Virtuelle Realität (Wahlpflichtfach)
Kennung	1211909
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Ziel der Veranstaltung ist die Vermittlung von grundlegenden Methoden zur Simulation virtueller Umgebungen. Außerdem werden VR-Anwendungen aus dem technisch-wissenschaftlichen und industriellen Umfeld vorgestellt. Die Vorlesung wird von praktischen Vorführungen begleitet und behandelt folgende Themen: Natur und Geschichte der VR, physiologische Aspekte des dreidimensionalen Sehens, VR-spezifische Themen der 3D-Computergraphik, stereoskopische Projektionen, Graphik-, Projektions- und Interaktionshardware, VR Displays, Erfassung menschlicher Bewegungen, Kollisionserkennung, 3D User Interfaces, VR-Anwendungen in Industrie und Forschung
Lernziele/Lernergebnisse	Kenntnisse: Beim erfolgreichen Abschluss des Moduls sollten die Studierenden Kenntnisse über folgende Themen haben: Charakteristische Merkmale der Virtuellen Realität, Basis-Methoden und Algorithmen der Virtuellen Realität, 3D Interaktion, Stereoskopische Projektionen, Effiziente Methoden der Kollisionserkennung, Motion Tracking   Fertigkeiten: Die Studierenden sollten in der Lage sein: Die Qualität von VR Interfaces systematisch zu bewerten und zu vergleichen, Selbständig VR-Schnittstellen und -Anwendungen konzeptionell zu entwickeln   Kompetenzen: Basierend auf dem Wissen und den Fähigkeiten sollten die Studierenden in der Lage sein: Die Potenziale, Herausforderungen und Einschränkungen von VR-Techniken zu analysieren und einzuordnen, und die erlernten Techniken und Methoden der Virtuellen Realität auf neue Problemstellungen zu übertragen
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Grundlagenwissen in linearer Algebra.
Literatur	D. Bowman et al. 3D User Interfaces. Addison-Wesley   K. M. Stanney. Handbook of Virtual Environments. Erlbaum   M. Slater et al. Computer Graphics & Virtual Environments. Addison-Wesley   G. Burdea, P. Coiffet. Virtual Reality Technology. John Wiley & Sons   K.-F. Kraiss (Ed.). Advanced Man Machine Interfaces. Springer   R.S. Kalawski. The Science of Virtual Reality and Virtual Environments. Addison Wesley
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Torsten Wolfgang Kuhlen
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)

– Applied Computer Science  
+ Virtuelle Realität (1211909)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Virtuelle Realität (121190902)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Virtuelle Realität (121190901)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Virtuelle Realität	1. Semester	keine Semesterempfehlung	-	3

Modultitel	Advanced Topics in Machine Learning for Human Language Technology (Wahlpflichtfach)
Kennung	1230525
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Sommersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Part 1: Probabilistic foundations, Bayes decision theory, probabilistic interpretation of neural networks, training criteria.</li> <li>• Part 2: Sequence processing and specific ANN structures (hidden markov models, finite-state transducers, cross-attention).</li> <li>• Part 3: Deep Learning and HLT tasks (automatic speech recognition, language modelling, machine translation).</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge:</p> <p>Today data-driven methods like machine learning and artificial neural networks (ANN) are widely used for speech and language processing, e.g. for automatic speech recognition (ASR) and machine translation. We will re-visit the evolution of these methods over the last 50 years and will present a unifying view of their principles from a probabilistic perspective.</p> <p>On successful completion of this module, students should be able to:</p> <ul style="list-style-type: none"> <li>• describe the components and formalisms of machine learning problems in human language technology from a principled probabilistic perspective</li> <li>• state and interpret the optimization problems of machine learning for human language technology and its relation to performance evaluation.</li> </ul> <p>Skills:</p> <p>They should be able to:</p> <ul style="list-style-type: none"> <li>• apply state-of-the-art machine learning models to human language technology components</li> <li>• solve the optimization problems underlying decoding in sequence classification based on state-of-the-art machine learning components and underlying models</li> <li>• should have acquired soft skills like acquiring, reproducing and discussing knowledge about current state-of-the-art machine learning models for human language technology in a cooperative environment.</li> </ul> <p>Competences:</p> <ul style="list-style-type: none"> <li>• understand the probabilistic interpretation of artificial neural network outputs</li> <li>• describe the relation between the task performance (e.g. word error rate in ASR) and the decision rule for generating the output sequence (e.g. Bayes decision rule)</li> <li>• relate training criteria (like cross-entropy) to task performance</li> <li>• be able to model the dependencies between input and output sequences in sequence-to-sequence processing</li> <li>• know the synchronization mechanisms between input and output sequences (e.g. hidden Markov models, finite-state transducers, cross-attention)</li> <li>• understand the role language modeling plays in the context of end-to-end models.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Das Modul "Advanced Statistical Classification" (1212684) darf nicht erfolgreich abgeschlossen sein.



– Applied Computer Science

+ Advanced Topics in Machine Learning for Human Language ...

(empfohlene) Voraussetzungen	Fundamentals of automatic speech recognition and machine learning, Statistical Classification and Machine Learning or further lectures on Machine Learning and/or Human Language Technology.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Prof. Dr.-Ing. Hermann Ney
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Advanced Topics in Machine Learning for Human Language Technology (123052501)	1. Semester	keine Semesterempfehlung	4	0
Übung Advanced Topics in Machine Learning for Human Language Technology (123052502)	1. Semester	keine Semesterempfehlung	0	1

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Advanced Topics in Machine Learning for Human Language Technology	1. Semester	keine Semesterempfehlung	-	2

Modultitel	High-performance Matrix Computations (Wahlpflichtfach)
Kennung	1211911
Version	V3
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	The course centers around the idea of developing efficient numerical algorithms for matrix computations through a synergy between mathematics and architectures. The focus is on the most common linear algebra operations: linear systems and eigenproblems. The objective is to attain high performance on a variety of parallel architectures: multi-core processors, GPUs, distributed and hybrid systems. Topics covered: generalized, standard and tridiagonal eigenproblems, matrix factorizations, linear systems. The students are expected to participate in practical programming exercises. The languages of choice are Matlab and C.
Lernziele/Lernergebnisse	Knowledge: On successful completion of this module, students will have knowledge about efficient numerical algorithms for linear systems and eigenproblems parallel architectures standard numerical linear algebra libraries: BLAS, LAPACK, MKL, Elemental performance metrics: efficiency, strong & weak scalability, time, space Skills: They should be able to write high-performance code for matrix operations in C design workqueue-based & multi-threaded parallel algorithms design message-passing-based parallel algorithms Competences: Based on the knowledge and skills acquired, they should be able to identify opportunities for parallelism in matrix operations tailor algorithms for a specific target architecture (shared memory architectures, distributed memory architectures, accelerators)
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Knowledge from Numerical linear algebra. Principles of algorithms and Knowledge from programming. Familiarity with Matlab and C.
Literatur	Lecture slides Diaries from the computer sessions G. Golub, C. van Loan. Matrix Computations. Third Edition, 1996. G. Meurant. Computer Solution of Large Linear Systems. North Holland, Amsterdam, 1999
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Universitätsprofessor Paolo Bientinesi Ph. D.
ECTS Credits	6
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	120

— Applied Computer Science

+ High-performance Matrix Computations (1211911)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	45,0
Selbststudium (h)	135,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Exam High-performance Matrix Computations (121191101)	1. Semester	keine Semesterempfehlung	6	0
Exercise High-performance Matrix Computations (121191102)	keine Semesterempfehlung	keine Semesterempfehlung	0	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture High-performance Matrix Computations	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Angewandte Softwaretechnik im Lebenszyklus der ...

Modultitel	Angewandte Softwaretechnik im Lebenszyklus der Automobilelektronik (Wahlpflichtfach)
Kennung	1215757
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2010
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Aufbau einer modernen Automobilelektronikarchitektur: Bussysteme, Steuergeräte, Software</li> <li>• Der Lebenszyklus einer Automobilelektronik</li> <li>• Vorgehen bei der Elektronikentwicklung</li> <li>• Kommunikation mit der Fahrzeugelektronik: Beispiel einer datengetriebenen Schichtenarchitektur - Einsatz internationaler Standards - Abstraktion vom konkreten Fahrzeug</li> <li>• Die Herausforderungen des Werkstattservice: Wie hilft die Softwaretechnik bei der Wartung der Fahrzeugelektronik von 20 Modellgenerationen - Variantenmanagement - Versions- / Konfigurationsmanagement - Datenlogistik</li> </ul>
Lernziele/Lernergebnisse	Mit diesem Modul wird ein praktischer Anwendungsbereich für moderne Softwaretechnik vermittelt: Die Automobilelektronik. Das Modul umfasst wesentliche moderne Techniken der heutigen Elektronikentwicklung und –wartung und hilft einerseits gelernte Methoden der Softwaretechnik in die Praxis zu applizieren und andererseits für einen eigenen Einstieg in automobile Themen vorbereitet zu sein.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Knowledge in Software Engineering
Literatur	<ul style="list-style-type: none"> <li>• W. Zimmermann, R. Schmidgal: Bussysteme in der Fahrzeugtechnik: Protokolle und Standards</li> <li>• J. Schäuffele, T. Zurawka: Automotive Software Engineering: Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen</li> <li>• Kai Borgeest: Elektronik in der Fahrzeugtechnik: Hardware, Software, Systeme und Projektmanagement</li> </ul>
Sprache	Deutsch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Dr. rer. nat. Ansgar SchleicherUniversitätsprofessor Dr. rer. nat. Bernhard Rumpke
ECTS Credits	2
Kontaktzeit (SWS)	1

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Angewandte Softwaretechnik im Lebenszyklus der ...

Prüfungsdauer (min)	0
Gesamtstunden (h)	60,0
Präsenzstunden (h)	15,0
Selbststudium (h)	45,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Masterprüfung Angewandte Softwaretechnik im Lebenszyklus der Automobilelektronik (121575701)	1. Semester	keine Semesterempfehlung	2	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Angewandte Softwaretechnik im Lebenszyklus der Automobilelektronik	1. Semester	keine Semesterempfehlung	-	1

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet ...

Modultitel	Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT) (Wahlpflichtfach)
Kennung	1215755
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	<ul style="list-style-type: none"> <li>• Aufbau einer modernen Automobilelektronikarchitektur: Bussysteme, Steuergeräte, Software</li> <li>• Der Lebenszyklus einer Automobilelektronik</li> <li>• Vorgehen bei der Elektronikentwicklung</li> <li>• Kommunikation mit der Fahrzeugelektronik: Beispiel einer datengetriebenen Schichtenarchitektur - Einsatz internationaler Standards - Abstraktion vom konkreten Fahrzeug</li> <li>• Die Herausforderungen des Werkstattservice: Wie hilft die Softwaretechnik bei der Wartung der Fahrzeugelektronik von 20 Modellgenerationen - Variantenmanagement - Versions- / Konfigurationsmanagement - Datenlogistik</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge</p> <ul style="list-style-type: none"> <li>• Mit diesem Modul wird ein praktischer Anwendungsbereich für moderne Softwaretechnik vermittelt: Die Automobilelektronik. Das Modul umfasst wesentliche moderne Techniken der heutigen Elektronikentwicklung und –wartung.</li> </ul> <p>Skills / Competences</p> <ul style="list-style-type: none"> <li>• Studierende sind nach Abschluss dieses Moduls in der Lage gelernte Methoden der Softwaretechnik in die Praxis zu applizieren</li> <li>• Sie sind für einen eigenen Einstieg in automobile Themen vorbereitet.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Kenntnisse aus dem Modul "Softwaretechnik".
Literatur	<ul style="list-style-type: none"> <li>• W. Zimmermann, R. Schmidgal: Bussysteme in der Fahrzeugtechnik: Protokolle und Standards</li> <li>• J. Schäuffele, T. Zurawka: Automotive Software Engineering: Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen</li> <li>• Kai Borgeest: Elektronik in der Fahrzeugtechnik: Hardware, Software, Systeme und Projektmanagement</li> </ul>
Sprache	Deutsch
Prüfungsbedingungen	Das Modul kann entweder mit 3 ECTS oder 6 ECTS abgeschlossen werden. Die Modulprüfung ist eine Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher:

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet ...

	Sebastiaan Wouters
	Modulverantwortlicher: Dr. rer. nat. Ansgar Schleicher; Universitätsprofessor Dr. rer. nat. Bernhard Rumpe
ECTS Credits	3
Kontaktzeit (SWS)	2
Prüfungsdauer (min)	-
Gesamtstunden (h)	90,0
Präsenzstunden (h)	30,0
Selbststudium (h)	60,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT) (121575504)	1. Semester	keine Semesterempfehlung	0	1
Praxisteil Programmierung (121575503)	1. Semester	keine Semesterempfehlung	0	0
Prüfung Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT) (3 Credits) (121575501)	1. Semester	keine Semesterempfehlung	3	0
Prüfung Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT) (6 Credits) (121575502)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Der digitale Lebenszyklus von Fahrzeugen als Teil des Internet of Things (IoT)	1. Semester	keine Semesterempfehlung	-	1

Modultitel	Digitalisierung (Wahlpflichtfach)
Kennung	1220230
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2019
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Die Vorlesung "Digitalisierung" beleuchtet das Thema Digitalisierung aus unterschiedlichen Perspektiven und Blickwinkeln:          Es werden Grundlagen zum Begriff Digitalisierung und seine Einordnung in die Fachdisziplin vermittelt. In Diskussionen wird die kritische Auseinandersetzung mit dem Begriff angeregt. Vortragende aus der Wirtschaft behandeln aktuelle Entwicklungen, die die Digitalisierung ihres Wirtschaftsbereichs voranbringen.          In einer schriftlichen Hausarbeit wird die Präsentation zu zwei Bereichen zusammengefasst und gegenübergestellt, um aktuelle Literatur ergänzt und im Bezug zur zukünftigen eigenen Karriere reflektiert.</p>
Lernziele/Lernergebnisse	<p>Einordnung des Begriffs Digitalisierung in der Fachdisziplin;          Kritische Auseinandersetzung mit dem Thema;          Kenntnisse über praktische Anwendungsgebiete;          Reflexion über die eigene Karriere.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	-
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Projektarbeit (100 %).
Sonstiges	-
Modulverantwortung	Prof. Dr. Bernhard Rumpe
ECTS Credits	3
Kontaktzeit (SWS)	1
Prüfungsdauer (min)	-
Gesamtstunden (h)	90,0
Präsenzstunden (h)	15,0
Selbststudium (h)	75,0



- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Digitalisierung (1220230)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Projektarbeit Digitalisierung (122023001)	2. Semester	keine Semesterempfehlung	3	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Digitalisierung	2. Semester	keine Semesterempfehlung	-	1

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Eingebettete Systeme (1215690)

Modultitel	Eingebettete Systeme (Wahlpflichtfach)
Kennung	1215690
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>Eingebettete Systeme steuern viele Dinge in unserem täglichen Leben. Energieeffiziente Kühlschränke, Aufzugssteuerungen und fortschrittliche Fahrerassistenzsysteme sind nur einige Beispiele. Embedded Systems steuern auch Prozesse im industriellen Umfeld und werden zur Erkennung und Vermeidung von Systemausfällen eingesetzt. Diese Vorlesung gibt eine allgemeine Einführung in das Thema Embedded Systems. Es werden grundlegende Konzepte vorgestellt und wichtige Unterschiede zu "normalen" Computersystemen aufgezeigt. Diese Vorlesung bereitet die Studierenden auf die Aufbauvorlesungen des Embedded Software Laboratory vor, die sich ausführlich mit Sicherheit, Zuverlässigkeit, formalen Methoden und dynamischen Systemen befassen. Diese Vorlesung richtet sich an alle Studierenden, die sich nicht nur auf das Verständnis von PCs beschränken wollen, sondern auch wissen wollen, wie z.B. Motorsteuergeräte und Produktionssteuerungssysteme funktionieren. Die in dieser Vorlesung behandelten Themen sind: Mikrocontroller, Speicherprogrammierbare Steuerungen (SPS, SPS-Programmiersprachen, Echtzeitanforderungen, Echtzeit-Betriebssysteme, Merkmale des Embedded-Software-Designs, Intra-Fahrzeugkommunikation (z.B. CAN-Bus), Teaser von Vorträgen des Embedded-Software-Labors. Die Vorlesung wird in deutscher Sprache mit englischen Folien gehalten.</p>
Lernziele/Lernergebnisse	<p>Kenntnisse: Kenntnisse und Vertrauen in moderne Softwaretechniken für eingebettete Systeme   Fertigkeiten: Fähigkeit, einen modellbasierten qualitätsorientierten Ansatz für das Design von Embedded Software   Kompetenzen anzuwenden: Sensibilität für besondere qualitative Anforderungen an das Design von Embedded Software.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der 'Grundlagen der Technischen Informatik'.
Literatur	Folien zur Vorlesung, Skript sowie als Ergänzung folgende Bücher: Marwedel: Eingebettete Systeme. 2003   Bass, Clements: Software Architecture in Practice.   Douglass: Real-time UML
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr.-Ing. Stefan Kowalewski
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Eingebettete Systeme (1215690)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Eingebettete Systeme (121569002)	1. Semester	keine Semesterempfehlung	0	1
Prüfung Eingebettete Systeme (121569001)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Eingebettete Systeme	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Mikrocontrollerprogrammierung und Fehlersuche (1220524)

Modultitel	Mikrocontrollerprogrammierung und Fehlersuche (Wahlpflichtfach)
Kennung	1220524
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2019
Gültig bis	-
Modulniveau	-
Inhalt	<p>The course aims to provide a comprehensive overview of methods for programming of microcontrollers and the debugging, respectively. There will be covered different programming techniques, backgrounds about the software development tool-chains best practices about the programming of microcontrollers and different techniques of de-bugging as deep as performing measurements at the copper layer.</p> <p>The course will dive deeper into the following topics:</p> <ul style="list-style-type: none"> <li>- Practical aspects of the compiler tool-chain</li> <li>- Programming language C</li> <li>- Coding guidelines</li> <li>- Runtime behavior</li> <li>- Debugging at the copper layer</li> <li>- Protective Circuits</li> <li>- First embedded insights to model-based software development and testing</li> </ul> <p>The above will be complemented with hands-on assignments using various examples (still to be determined).</p>
Lernziele/Lernergebnisse	After the course student should have a good overview of programming a microcontroller in an embedded environment in a structured manner. The interaction between the microcontroller as device under development and the embedding environment should be understood in a way that debugging malfunctions is fostered.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	To be decided (several books are being reviewed). Probably a collection of book chapters and articles covering the data science spectrum will be used, complemented by slides.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100%). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Dr. Ing. André Stollenwerk
ECTS Credits	6
Kontaktzeit (SWS)	6
Prüfungsdauer (min)	-

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Mikrocontrollerprogrammierung und Fehlersuche (1220524)

Gesamtstunden (h)	180,0
Präsenzstunden (h)	90,0
Selbststudium (h)	90,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Mikrocontrollerprogrammierung und Fehlersuche (Prüfung) (122052401)	2. Semester	keine Semesterempfehlung	6	0
Mikrocontrollerprogrammierung und Fehlersuche (Übung) (122052402)	2. Semester	keine Semesterempfehlung	0	2

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Mikrocontrollerprogrammierung und Fehlersuche (Vorlesung)	2. Semester	keine Semesterempfehlung	-	4

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Simulation Software Engineering (8015429)

Modultitel	Simulation Software Engineering (Wahlpflichtfach)
Kennung	8015429
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester
Gültig von	Wintersemester 2009
Gültig bis	-
Modulniveau	Master
Inhalt	<p>Diese Vorlesung behandelt Softwareengineering-Techniken und -Werkzeuge für die Entwicklung paralleler Simulationscodes.</p> <ul style="list-style-type: none"> <li>- Softwareentwicklungsprozess und -lebenszyklus</li> <li>- Anforderungsanalyse</li> <li>- Entwurf</li> <li>- Qualitätssicherung</li> <li>- Einsatz und Pflege</li> <li>- Wissenschaftliche Bibliotheken</li> </ul>
Lernziele/Lernergebnisse	Die Studierenden sollen in die Lage versetzt werden, korrekte, robuste, effiziente und wartbare parallele numerische Software zu schreiben.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Programmiersprache C/C++.
Literatur	<p>I. Sommerville: Software Engineering, Addison-Wesley, 8th edition, 2006.</p> <p>S. Oliveira, D. Stewart: Writing Scientific Software, Cambridge University Press, 2006</p>
Sprache	Englisch
Prüfungsbedingungen	<p>Die Benotung ergibt sich aus 3 Teilen:</p> <ul style="list-style-type: none"> <li>• Zwei Referate welche jeweils mit 30% gewichtet werden;</li> <li>• Eine schriftliche Hausarbeit welche mit 40% gewichtet wird.</li> </ul>
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Felix Gerd Eugen Wolf
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Simulation Software Engineering (8015429)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Masterprüfung Simulation Software Engineering (801542901)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Simulation Software Engineering	1. Semester	keine Semesterempfehlung	-	2
Vorlesung Simulation Software Engineering	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Software-Projektmanagement (1212355)

Modultitel	Software-Projektmanagement (Wahlpflichtfach)
Kennung	1212355
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Die folgenden Themen werden behandelt: Grundlagen des Projektmanagements und zentrale Prozesse   Prozessmodelle der Softwareentwicklung   Agile Entwicklungsmodelle   Techniken der Projektplanung   Klassische und agile Techniken zur Kostenschätzung   Techniken der Zeitplanung   Projekt-Monitoring und Controlling   Risikomanagement   Stakeholder-Management und Entscheidungstechniken
Lernziele/Lernergebnisse	Nach Beendigung des Moduls verfügen die Studierenden über das folgende Wissen und über die folgenden Kompetenzen: Sie kennen die Ziele, wichtige Aktivitäten und Techniken des Software Projektmanagements   sie wissen, wie Entwicklungsaktivitäten und Projektmanagement Aktivitäten zusammenarbeiten   sie sind in der Lage kleinere Projekte zu planen, Aufwandsabschätzungen durchzuführen, den Projektfortschritt festzustellen und zu überwachen und systematisch Risiken zu bewerten   sie können Stakeholder analysieren und verfügen über Techniken, die helfen, Entscheidungen systematisch zu treffen. Nutzen für die zukünftige Karriere / vermittelte Soft Skills: Alle Kompetenzen werden im Rahmen der begleitenden Übungen trainiert. In diesen arbeiten die Studierenden in Gruppen zusammen, um ihre Lösungen zu entwickeln. Weiterhin präsentieren und diskutieren sie im Rahmen der Übungen die von ihnen entwickelten Lösungen. Da in diesem Modul professionellen Wissen im Bereich des Projektmanagements vermittelt wird, verfügen die Studierenden am Ende über relevantes Wissen und Kompetenzen, um erfolgreich in Führungspositionen in der Industrie arbeiten zu können.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Softwaretechnik.
Literatur	H. Kerzner (2002): Project Management - A Systems Approach to Planning, Scheduling and Controlling, Wiley.   IEEE Std 1490 (2003): Adoption of PMI Standard A Guide to the Project Management Body of Knowledge   Wysocki, R. (2007): Effective Project Management, Wiley Publishing Inc.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Horst Lichter
ECTS Credits	4
Kontaktzeit (SWS)	3
Prüfungsdauer (min)	15-45 (mündlich/oral)   60-90 (schriftlich/written)



- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Software-Projektmanagement (1212355)

Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Software-Projektmanagement (121235501)	1. Semester	keine Semesterempfehlung	4	0
Übung Software-Projektmanagement (121235502)	1. Semester	keine Semesterempfehlung	0	1

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Software-Projektmanagement	1. Semester	keine Semesterempfehlung	-	2

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Software-Qualitätssicherung (1212356)

Modultitel	Software-Qualitätssicherung (Wahlpflichtfach)
Kennung	1212356
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>This module introduces central concepts, methods, techniques and processes of software quality-assurance.</p> <p>The following topics are covered:</p> <ul style="list-style-type: none"> <li>- Terms, concepts and models of quality assurance</li> <li>- Measurement and software metrics</li> <li>- Quality models</li> <li>- Test automation</li> <li>- Foundations of tests and test theory</li> <li>- Test techniques, section of test cases</li> <li>- Test-driven Development and Behavior-driven Development</li> <li>- Approaches to static examination of software</li> <li>- Foundations on metrics and measurement</li> <li>- Economic models of quality assurance</li> </ul>
Lernziele/Lernergebnisse	<p>General:</p> <p>After completing the module the students have the following knowledge and competencies. They ...</p> <ul style="list-style-type: none"> <li>- know the goals, concepts, models, and basic terms of software quality assurance</li> <li>- know important methods of static software inspections</li> <li>- are able to apply test case selection techniques and know important test exit criteria</li> <li>- are able to systematically develop test specifications</li> <li>- know the fundamentals of software measurement and are able to define and assess software metrics</li> <li>- know standard approaches to evaluate and improve software development processes</li> </ul> <p>Benefits for future professional life / soft skills:</p> <p>All competencies are trained in the exercises, where small teams of students have to create typical software quality assurance artifacts. They have to present and discuss their solutions and ideas in front of the class. As professional knowledge on software quality assurance is provided, students gain personal and professional competencies that enable to work as quality assurance engineer.</p>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Learning outcomes for the Module 'Software Engineering'
Literatur	<p>Spillner, A., Linz, T., &amp; Schaefer, H. (2006): Software Testing Foundations - A Study Guide for the Certified Tester Exam. dpunkt.verlag Heidelberg. Paul C. Jorgensen (2013): Software Testing: A Craftsman's Approach (4th ed.). Auerbach Publications, Boston, MA, USA. Michal Young and Mauro Pezze (2005): Software Testing and Analysis: Process, Principles and Techniques. John Wiley &amp; Sons.</p>
Sprache	Englisch

- Software Engineering
- Wahlpflichtbereich Software Engineering
- + Software-Qualitätssicherung (1212356)

Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor Dr. rer. nat. Horst Lichter
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Software- Qualitätssicherung (121235602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Software- Qualitätssicherung (121235601)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Software- Qualitätssicherung	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Core Subjects Software Engineering
- + Designing Interactive Systems I (1215698)

Modultitel	Designing Interactive Systems I (Wahlpflichtfach)
Kennung	1215698
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Dieser Kurs führt die Schüler in die Mensch-Computer-Interaktion (HCI) und das Design von Benutzeroberflächen ein. Es werden die folgenden Themen behandelt: Grundlegende Merkmale der menschlichen Wahrnehmung wie Reaktionszeit, Wahrnehmungsregeln und Gedächtnisleistung, Modelle der Interaktion zwischen Menschen und ihrer Umgebung wie Leistungen, Abbildungen, Einschränkungen, Ausrutscher und Fehler. Außerdem gibt es eine Einführung in die Geschichte und Grundlagen der Mensch-Computer-Interaktion, Prinzipien des iterativen Designs, Prototypentechniken für Benutzeroberflächen, Goldene Regeln für das Design von Benutzeroberflächen, sowie Richtlinien für das Design von Benutzeroberflächen, Benutzerstudien und Bewertungsmethoden.
Lernziele/Lernergebnisse	Wissen: Nach diesem Kurs wissen die Schüler, wie sich Benutzeroberflächen in den letzten Jahrzehnten entwickelt haben und welche Eigenschaften der Menschen bei der Gestaltung berücksichtigt werden müssen.   Fähigkeiten: Sie werden in der Lage sein, iteratives Design, Prototyping und Evaluierungsmethoden anzuwenden, um benutzerzentriert verwendbare, geeignete Benutzeroberflächen zu entwerfen. Alle Aufgaben sind Gruppenaufgaben zur Förderung der Kollaborationsfähigkeiten und projektbasiert, um Projektplanung, Konfliktmanagement und Präsentationsfähigkeiten zu stärken.   Kompetenzen: Die Studierenden lernen, in Designer-Begriffen zu denken. Dies ist eine entscheidende Kompetenz für Informatiker, die an Benutzeroberflächen arbeiten und erfordert die Zusammenarbeit in interdisziplinären Teams als Vorbereitung für den späteren Berufsalltag.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine. ;
Literatur	D. Norman: The Design Of Everyday Things, Basic Books 2002 (verpflichtend für die ersten Wochen des Kurses), plus Auszüge aus A. Dix et al.: Human-Computer Interaction, Prentice-Hall 2004. Shneiderman et al.: Designing The User Interf. Add.-W. 2004, J. Raskin: The Humane Interface, Addison-Wesley, 2000
Sprache	Englisch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Projektarbeit mit Referat (40 %); Klausur oder mündliche Prüfung (60 %). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Jan Oliver Borchers
ECTS Credits	6
Kontaktzeit (SWS)	5

- Software Engineering
- Core Subjects Software Engineering
- + Designing Interactive Systems I (1215698)

Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Designing Interactive Systems I (121569802)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Designing Interactive Systems I (121569801)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Designing Interactive Systems I	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Core Subjects Software Engineering
- + Formale Methoden für Steuerungssoftware (1212666)

Modultitel	Formale Methoden für Steuerungssoftware (Wahlpflichtfach)
Kennung	1212666
Version	V2
Dauer (Semester)	-
Turnus (Semester)	-
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	-
Inhalt	<p>Die Vorlesung wird eine anwendungsorientierte Einführung in verschiedene Formale Methoden geben. Dabei liegt der Fokus auf dem Einsatz dieser bei der Entwicklung von Steuerungssoftware. Die Themen der Vorlesung umfassen:</p> <ul style="list-style-type: none"> <li>• Statische Analyse</li> <li>• Abstract Interpretation</li> <li>• Program Slicing</li> <li>• Spezifikationen</li> <li>• Model-Checking</li> <li>• Concolic Testing</li> </ul>
Lernziele/Lernergebnisse	<p>Wissen: Nach Abschluss des Moduls werden die Studierenden die folgenden Punkte erklären können:</p> <ul style="list-style-type: none"> <li>• Herausforderungen bei der Analyse von Steuerungssoftware</li> <li>• Techniken und Formalismen zur statischen Analyse</li> <li>• Vor- und Nachteile verschiedener abstrakter Domänen</li> <li>• Methoden und Anwendungen von Program Slicing</li> <li>• Modellierung von SPS-Programmen zwecks Verifikation</li> <li>• Verschiedene Techniken zur Spezifikation von Systemanforderungen</li> <li>• Verfahren zur Verifikation verschiedener Spezifikationen</li> <li>• Techniken zur Kodierung von SPS-Semantik als Formeln für SMT-Solver</li> </ul> <p>Fähigkeiten: Die Studierenden sollten in der Lage sein, die folgenden Techniken zu erklären und anzuwenden:</p> <ul style="list-style-type: none"> <li>• Aufbau eines Frameworks für statische Analyse</li> <li>• Ableiten verschiedener Programmeigenschaften durch die Wahl geeigneter Abstrakter Domänen für die statische Analyse</li> <li>• Abhängigkeitsanalyse und Vereinfachung durch Program Slicing</li> <li>• Kodierung von SPS-Programmen als Formeln zur Erfüllbarkeitsüberprüfung mit SMT-Solvern</li> <li>• Spezifikationen von Anforderungen durch geeignete Formalisierungen entwerfen</li> <li>• Überprüfen von Spezifikationen mittels geeigneter Model-Checking Algorithmen</li> </ul> <p>Kompetenzen: Basierend auf dem gewonnenen Wissen und Fähigkeiten sollten die Studierenden in der Lage sein</p> <ul style="list-style-type: none"> <li>• Eine statische Analyse für Steuerungssoftware mit geeigneten abstrakten Domänen zu entwickeln,</li> <li>• Programme und Anforderungen zu modellieren, bzw. spezifizieren und</li> <li>• Geeignete Techniken aus einem Fundus Formaler Methoden auf Problemstellungen im Bereich der Programmanalyse anzuwenden.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Vorkenntnisse in statischer Analyse und/oder Modellüberprüfung sind von Vorteil, aber nicht zwingend erforderlich.
Literatur	-
Sprache	Englisch

- Software Engineering
- Core Subjects Software Engineering
- + Formale Methoden für Steuerungssoftware (1212666)

Prüfungsbedingungen	Klausur (100 %).
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Universitätsprofessor Dr.-Ing. Stefan Kowalewski Hendrik Simon M. Sc. RWTH
ECTS Credits	6
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	60,0
Selbststudium (h)	120,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Formale Methoden für Steuerungssoftware (121266602)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Formale Methoden für Steuerungssoftware (121266601)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Formale Methoden für Steuerungssoftware	1. Semester	keine Semesterempfehlung	-	2

- Software Engineering
- Core Subjects Software Engineering
- + Modellbasierte Softwareentwicklung (1215686)

Modultitel	Modellbasierte Softwareentwicklung (Wahlpflichtfach)
Kennung	1215686
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	Nach einer grundlegenden Einführung in die UML werden die Verwendungsmöglichkeiten von Modellen im Softwareentwicklungsprozess diskutiert. Dazu gehören Simulation, Code- und Test-Fallgenerierung, Analyse von Modellen und Evolution von Systemen durch Refactoring von Modellen.
Lernziele/Lernergebnisse	Kenntnisse: UML, Verwendung von Modellen im Softwareentwicklungsprozess, Simulation und Generierung von Code und Testfällen aus Modellen, Analyse von Modellen, Evolution von Modellen durch Refactoring.   Fertigkeiten: Anwendung von Modellen im Entwicklungsprozess.   Kompetenzen: Verständnis des Nutzen von Modellen, Verständnis und Anwendung der UML.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Softwaretechnik.
Literatur	B. Rumpe: Modellierung mit UML: Sprache, Konzepte und Methodik, Springer, Mai 2004; B. Rumpe : Agile Modellierung mit UML : Codegenerierung, Testfälle, Refactoring. Springer, August 2004
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Bernhard Rumpe
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0



- Software Engineering
- Core Subjects Software Engineering
- + Modellbasierte Softwareentwicklung (1215686)

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Modellbasierte Softwareentwicklung (121568602)	1. Semester	keine Semesterempfehlung	0	3
Prüfung Modellbasierte Softwareentwicklung (121568601)	1. Semester	keine Semesterempfehlung	6	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Modellbasierte Softwareentwicklung	1. Semester	keine Semesterempfehlung	-	2

- Software Engineering
- Core Subjects Software Engineering
- + Objektorientierte Softwarekonstruktion (1212354)

Modultitel	Objektorientierte Softwarekonstruktion (Wahlpflichtfach)
Kennung	1212354
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Master
Inhalt	Dieses Modul führt in zentrale Methoden, Techniken und Prozesse einer systematischen Softwareentwicklung basierend auf objekt-orientierten Konzepten ein. Die Vorlesung beschäftigt sich mit den folgenden Themen: Grundlagen der Objektorientierung   Software Wiederverwendung   Design by Contract   Vererbung, Polymorphismus und Generische Einheiten   Software Entwurfsprinzipien   Domain Modelling, Domain Driven Design   Komponententechnologie   Muster und Rahmenwerke   Smells und Refactoring
Lernziele/Lernergebnisse	Nach Beendigung des Moduls verfügen die Studierenden über das folgende Wissen und über die folgenden Kompetenzen: sie wissen, wie objekt-orientierte Modellierungskonzepte anzuwenden sind   sie sind in der Lage Use-Case basiert zu analysieren und Domänenwissen zu modellieren   sie kennen die Konzepte zur Entwicklung von Rahmenwerken   sie sind in der Lage, Code und Architektur auf Basis von Smells zu verbessern   sie kennen die Architektur von Java basierten Komponentenmodellen. Nutzen für die zukünftige Karriere / vermittelte Soft Skills: Alle Kompetenzen werden im Rahmen der begleitenden Übungen trainiert. In diesen arbeiten die Studierenden in Gruppen zusammen, um ihre Lösungen zu entwickeln. Weiterhin präsentieren und diskutieren sie im Rahmen der Übungen die von ihnen entwickelten Lösungen. Da in diesem Modul professionelles Wissen im Bereich der systematischen Softwarekonstruktion vermittelt wird, verfügen die Studierenden am Ende über relevantes Wissen und Kompetenzen um als Softwarearchitekt erfolgreich arbeiten zu können.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse der Softwaretechnik.
Literatur	Meyer, B. (1997): Object Oriented Software Construction, 2nd edition, Prentice Hall   Züllighoven, H. (2005): Object-Oriented Construction Handbook – Developing Application-Oriented Software with the Tools and Materials Approach. dpunkt.verlag, Heidelberg   Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (1995): Design Patterns, Addison-Wesley   Fowler Martin (1999): Refactoring - Improving the design of existing code, Addison Wesley.
Sprache	Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. rer. nat. Horst Lichter
ECTS Credits	6
Kontaktzeit (SWS)	5

- Software Engineering
- Core Subjects Software Engineering
- + Objektorientierte Softwarekonstruktion (1212354)

Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Objekt-orientierte Softwarekonstruktion (121235402)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Objekt-orientierte Softwarekonstruktion (121235401)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Objekt-orientierte Softwarekonstruktion	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Core Subjects Software Engineering
- + Software Language Engineering (1216957)

Modultitel	Software Language Engineering (Wahlpflichtfach)
Kennung	1216957
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<p>„Die Grenzen meiner Sprache bedeuten die Grenzen meiner Welt.“ Ludwig Wittgenstein. Dies trifft umso mehr zu, wenn Menschen mit Computern kommunizieren.</p> <p>Der Kurs beschäftigt sich mit den Konzepten zur Sprachdefinition, wie etwa Metamodellen, Grammatiken, modernen Editoren und zur Verwendung von Software Sprachen, zum Beispiel zur Modellierung von Software, Systemen, Simulationen. Dabei werden Beispiele, wie etwa die UML, domänenspezifische Sprachen (DSL) und XML diskutiert und semantische Analyse und Generierungstechniken besprochen.</p> <p>DSLs eignen sich immer da, wo Nicht-Informatiker mit Computern zu tun haben und komplexe Sachverhalte modellhaft beschreiben müssen. Modelle in kompakten DSL's eignen sich z.B. hervorragend zu Generierung und zur Konfiguration von Systemen, zur Orchestrierung von Webservices oder auch zur Ablaufsteuerung von Geschäftsprozessen. Sie eignen sich zur Modellierung des Gehirns ebenso wie von autonomen Fahrzeugen oder E-Home Geräten.</p>
Lernziele/Lernergebnisse	<p>Knowledge</p> <ul style="list-style-type: none"> <li>• Domänenspezifische Sprachen</li> <li>• Generator-Technologie</li> <li>• Simulation und Generierung aus Modellen</li> <li>• Analyse von Modellen</li> </ul> <p>Skills</p> <ul style="list-style-type: none"> <li>• Entwicklung von DSLs</li> <li>• Verständnis zur Nutzung einer Language Workbench</li> <li>• Verständnis des Nutzens von DSLs für Entwicklung und Simulation von Systemen</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Kenntnisse aus Einführung in die Softwaretechnik.
Literatur	<ul style="list-style-type: none"> <li>• [Rum17] B. Rumpe: Agile Modeling with UML: Code Generation, Testing, Refactoring. Springer International, May 2017.</li> <li>• [CFJ+16] B. Combemale, R. France, J. Jézéquel, B. Rumpe, J. Steel, D. Vojtisek: Engineering Modeling Languages: Turning Domain Knowledge into Tools. Chapman &amp; Hall/CRC Innovations in Software Engineering and Software Development Series, November 2016.</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Projektarbeit (100 %).
Sonstiges	-
Modulverantwortung	<p>Modulangebotsorganisator:</p> <p>Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher:</p> <p>Dr. rer. nat. Katja PetzoldtModulverantwortlicher:</p>

- Software Engineering
- Core Subjects Software Engineering
- + Software Language Engineering (1216957)

	Universitätsprofessor Dr. rer. nat. Bernhard Rumpe
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Software Language Engineering (121695702)	1. Semester	keine Semesterempfehlung	0	3
Prüfung Software Language Engineering (121695701)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Software Language Engineering	1. Semester	keine Semesterempfehlung	-	2

- Software Engineering
- Core Subjects Software Engineering
- + Software-Architekturen (1215687)

Modultitel	Software-Architekturen (Wahlpflichtfach)
Kennung	1215687
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2018
Gültig bis	-
Modulniveau	Bachelor/Master
Inhalt	<ul style="list-style-type: none"> <li>Modeling at design level</li> <li>A module concept</li> <li>Subarchitectures and extensions of the module concept</li> <li>Transformation into programming languages</li> <li>Architecture examples</li> <li>Strategies for adaptability and reusability</li> <li>Expressing semantics</li> <li>Expressing distribution</li> <li>Concurrent &amp; embedded systems</li> <li>Concrete and abstract component connections</li> </ul>
Lernziele/Lernergebnisse	<p>Knowledge</p> <ul style="list-style-type: none"> <li>role of design in development processes</li> <li>types of small or large components</li> <li>range of relations between components in architectures</li> <li>annotations for different purposes</li> </ul> <p>Skills</p> <ul style="list-style-type: none"> <li>building up small architectures</li> <li>applying rules and patterns</li> <li>translate to programming languages</li> <li>find situations where to apply data abstraction/ OO</li> </ul> <p>Competences</p> <ul style="list-style-type: none"> <li>evaluate an architecture due to adaptability, portability, and reuse</li> <li>overviewing the architectures of batch, interactive, and embedded systems</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Knowledge of 'Introduction to Software Engineering'

- Software Engineering
- Core Subjects Software Engineering
- + Software-Architekturen (1215687)

Literatur	<ul style="list-style-type: none"> <li>• M. Nagl: Methodisches Programmieren 1990</li> <li>• weitere schriftliche Unterlagen</li> <li>• andere Lehrbücher zur Ergänzung</li> </ul>
Sprache	Deutsch/Englisch
Prüfungsbedingungen	Klausur oder mündliche Prüfung (100 %). Voraussetzung für die Zulassung zur Modulprüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher InformatikModellierungsteamverantwortlicher: Dr. rer. nat. Katja PetzoldtModulverantwortlicher: Universitätsprofessor i.R. Dr.-Ing. Manfred Nagl
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	0
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Übung Software-Architekturen (121568702)	1. Semester	keine Semesterempfehlung	0	2
Prüfung Software-Architekturen (121568701)	1. Semester	keine Semesterempfehlung	6	0

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Software-Architekturen	1. Semester	keine Semesterempfehlung	-	3

- Software Engineering
- Core Subjects Software Engineering
- + Model-based Systems Engineering (1222882)

Modultitel	Model-based Systems Engineering (Wahlpflichtfach)
Kennung	1222882
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	After a thorough and detailed introduction of SysML and UML, the possibilities of using models in system development processes are discussed. These include simulation, code and test case generation, analysis, modeling and evolution of systems by refactoring of models.
Lernziele/Lernergebnisse	<p>Knowledge:</p> <ul style="list-style-type: none"> <li>• SysML, UML</li> <li>• MontiArc</li> <li>• Architecture and behavior models</li> <li>• Statecharts, finite automata</li> <li>• Object diagrams and class diagrams</li> <li>• Geometrical models and their connection to software controlling models</li> <li>• Use of models in the software and systems engineering process</li> <li>• Simulation, code, and test generation</li> <li>• Analysis of models</li> <li>• Evolution of models and systems</li> </ul> <p>Skills:</p> <ul style="list-style-type: none"> <li>• Application of models in the development process</li> <li>• Ability to read and write own models in appropriate languages</li> </ul> <p>Competences:</p> <ul style="list-style-type: none"> <li>• Understanding of the use of models</li> <li>• Application of models in software and systems engineering</li> <li>• Knowledge and practice of SysML and UML</li> <li>• Designing systems with a strong software impact by using model-based development techniques</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	Keine
(empfohlene) Voraussetzungen	Introduction to software engineering or comparable courses.
Literatur	<p>[Rum17] B. Rumpe: Agile Modeling with UML: Code Generation, Testing, Refactoring. Springer International, May 2017.</p> <p>[CFJ+16] B. Combemale, R. France, J. Jézéquel, B. Rumpe, J. Steel, D. Vojtisek: Engineering Modeling Languages: Turning Domain Knowledge into Tools. Chapman &amp;; Hall/CRC Innovations in Software Engineering and Software Development Series, November 2016.</p>
Sprache	Englisch
Prüfungsbedingungen	Projektarbeit (100%). Voraussetzung für die Zulassung zur Prüfung ist das Bestehen von Hausaufgaben.
Sonstiges	-



- Software Engineering
- Core Subjects Software Engineering
- + Model-based Systems Engineering (1222882)

Modulverantwortung	-
ECTS Credits	6
Kontaktzeit (SWS)	5
Prüfungsdauer (min)	-
Gesamtstunden (h)	180,0
Präsenzstunden (h)	75,0
Selbststudium (h)	105,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Model-based Systems Engineering (122288201)	1. Semester	keine Semesterempfehlung	6	0
Übung Model-based Systems Engineering (122288202)	1. Semester	keine Semesterempfehlung	0	3

### ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Vorlesung Model-based Systems Engineering	1. Semester	keine Semesterempfehlung	-	2

## + Schwerpunktkolloquium (1222540)

Modultitel	Schwerpunktkolloquium (Wahlpflichtfach)
Kennung	1222540
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	The depth oral is an oral examinations about modules with a total of 12-18 CP. The subject of these modules should be closely related and it is possible that some or all of these modules are also part of the electives taken by the student.
Lernziele/Lernergebnisse	Knowledge: The student gains in-depth knowledge of a particular area of Computer Science as covered by the modules chosen for this examination. In addition to the knowledge each individual module offers, the student also gains insights about the connections between these modules. Skills: This modules enables to student to engage in a scientific discussion about a specific area of Computer Science draw connections between closely related sub-areas of Computer Science Competences: The student is able to study and comprehend an area of Computer Science in a larger context
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	None.
Literatur	Course material from the modules covered in the depth oral.
Sprache	Englisch
Prüfungsbedingungen	Mündliche Prüfung (100 %).
Sonstiges	-
Modulverantwortung	Modulangebotsorganisator: Modulangebotsverantwortlicher Informatik Modellierungsteamverantwortlicher: Dr. rer. nat. Katja Petzoldt Modulverantwortlicher: Fachgruppe Informatik
ECTS Credits	3
Kontaktzeit (SWS)	0
Prüfungsdauer (min)	-
Gesamtstunden (h)	90,0
Präsenzstunden (h)	,0
Selbststudium (h)	90,0

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Schwerpunktkolloquium (122254001)	keine Semesterempfehlung	keine Semesterempfehlung	3	0

+ Praktikum (1215759)

Modultitel	Praktikum (Wahlpflichtfach)
Kennung	1215759
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2009
Gültig bis	-
Modulniveau	Master
Inhalt	Im Praktikum sollen die Studierenden selbstständig fachspezifische Kenntnisse und Methoden bei der Konzeption, der Implementierung und dem Test von Software- und Hardware-Systemen sowie bei der Durchführung von Experimenten und Messungen anwenden. Üblicherweise erfolgt die Bearbeitung einer Aufgabenstellung in Kleingruppen.
Lernziele/Lernergebnisse	Kenntnisse: Nach erfolgreichem Abschluss dieses Moduls sollten die Studierenden in der Lage sein, die Vor- und Nachteile verschiedener Methoden zur Problemlösung für die jeweilige Anwendungsdomäne darzustellen.   Fertigkeiten: Sie sollten in der Lage sein, Ideen zu sammeln und zu filtern und iterativ Anwendungen zu einem bestimmten Thema zu entwickeln.   Kompetenzen: Basierend auf den erworbenen Kenntnissen und Fertigkeiten sollten sie in der Lage sein, mündlich zu kommunizieren, in Teams zu arbeiten, Projekte zu planen und Meilensteine zu erreichen.
Teilnahmebedingungen (studiengangspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Praktikum (100 %). Im Praktikum besteht Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Fachgruppe Informatik
ECTS Credits	7
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	-
Gesamtstunden (h)	210,0
Präsenzstunden (h)	60,0
Selbststudium (h)	150,0

+ Praktikum (1215759)

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Praktikum (121575901)	3. Semester	keine Semesterempfehlung	7	0

▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Praktikum	3. Semester	keine Semesterempfehlung	-	4

+ Seminar Informatik (1211974)

Modultitel	Seminar Informatik (Wahlpflichtfach)
Kennung	1211974
Version	V2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Sommersemester 2020
Gültig bis	-
Modulniveau	Bachelor
Inhalt	Das Erreichen der Lernziele wird durch Einübung an Hand persönlich zugeordneter vertiefter wissenschaftlicher Themen sowie die aktive Teilnahme an den Präsentationsterminen verfolgt. Die Wahl der Themengebiete obliegt dem jeweiligen Veranstalter.
Lernziele/Lernergebnisse	Kenntnisse: Ausgewählte fortgeschrittene Themen der Informatik und Präsentationstechniken.   Fähigkeiten: Auf der Basis geeigneter Literatur, insbesondere wissenschaftlicher Originalartikel, eigenständig in ein fortgeschrittenes Thema der Informatik einarbeiten, das Thema geeignet einordnen und eingrenzen sowie eine kritische Bewertung entwickeln. Konzepte, Vorgehensweisen und Ergebnisse eines vorgegebenen Themas der Informatik anschaulich und mit angemessenen Formalismen termingerecht und in definiertem Umfang vertieft schriftlich ausarbeiten; Nachweis der eigenständigen Erarbeitung durch Darstellung selbst gewählter Beispiele. Anschauliche mündliche Präsentation eines vertieften Themas der Informatik unter Einsatz geeigneter Medien und Beispiele in vorgegebener Dauer planen und durchzuführen. Aktive Teilnahme an Diskussionen zu vertieften Themen der Informatik in Präsenzveranstaltungen.   Kompetenzen: Konzepte, Vorgehensweisen und Ergebnisse eines wissenschaftlichen Themas der Informatik aufbereiten und präsentieren.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	Themenabhängig; wird vorgegeben bzw. selbst recherchiert.
Sprache	Deutsch
Prüfungsbedingungen	Schriftliche Hausarbeit mit Referat (100 %). Im Seminar besteht Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Fachgruppe Informatik
ECTS Credits	4
Kontaktzeit (SWS)	2
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	30,0
Selbststudium (h)	90,0

+ Seminar Informatik (1211974)

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Seminar (121197401)	2. Semester	keine Semesterempfehlung	4	2

— Seminar aus anderen Fakultäten/Fachgruppen

+ Seminar: Mathematical Concepts of Machine Learning (4028624)

Modultitel	Seminar: Mathematical Concepts of Machine Learning (Wahlpflichtfach)
Kennung	4028624
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Sommersemester
Gültig von	Sommersemester 2023
Gültig bis	-
Modulniveau	Master
Inhalt	<p>The use of machine learning and artificial intelligence has a huge potential for building future engineering systems such as autonomous robots, vehicles, or smart infra-structure systems. This is an interdisciplinary seminar offered in mechanical engineering and computer science due to the interdisciplinary nature of the topic. In addition to the challenges of designing ML and AI systems, it is critical to analyze the mathematical properties of these algorithms. This seminar will focus on some of the underlying mathematics of modern methods.</p> <p>The number of places for this seminar is limited.</p>
Lernziele/Lernergebnisse	<p><u>Wissen und Verstehen:</u></p> <p>You will learn about topics of current research in machine learning and focus on the underlying mathematical concepts. Many modern machine learning methods are readily implemented and can be used rather straightforwardly on a given data set. However, often this yields poor results. Thus, it is important to understand how the algorithms work and what mathematical properties of the data are critical to ensure good performance. A prominent example is correlated data, which can have dramatic consequences for certain methods and be manageable for others. The amount and quality of data is also an important aspect that heavily influences the success of the learning outcome.</p> <p><u>Fertigkeiten und Kompetenzen:</u></p> <p>You will prepare an assigned topic, familiarize with the mathematical details, and give a presentation to your peers. There will be theorems and proofs. You need to understand and reproduce the mathematical arguments, reasoning and rigor.</p>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	<p>Empfohlene Voraussetzungen:</p> <p>Prior knowledge in probability theory and machine learning is recommended.</p>
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	<p>The attendance of the course is mandatory.</p> <p>Presentation (40%) and written report (60%).</p>
Sonstiges	-
Modulverantwortung	Universitätsprofessor Dr. Sebastian Trimpe
ECTS Credits	4



— Seminar aus anderen Fakultäten/Fachgruppen

+ Seminar: Mathematical Concepts of Machine Learning (4028624)

Kontaktzeit (SWS)	3
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	45,0
Selbststudium (h)	75,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Seminar: Mathematical Concepts of Machine Learning (402862401)	2. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Seminar: Mathematical Concepts of Machine Learning	2. Semester	keine Semesterempfehlung	-	3

Modultitel	Current Topics in Applied Computational Social Sciences (Wahlpflichtfach)
Kennung	7023944
Version	V1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Unregelmäßig
Gültig von	Wintersemester 2020
Gültig bis	-
Modulniveau	Master
Inhalt	Applied Computational Social Science (ACSS) is an emerging field that focuses on the quality, validity and reliability of new types of data, data collection methods and computational data analysis methods for the social sciences. Beside methodological challenges that arise when collecting and analyzing digital behavioral data to build models of human behavior, attitudes and characteristics, the impact of digitalization on society and science is investigated. In this seminar students will review ACSS papers about one selected topical area and critically reflect on quality, validity and reliability of new types of data, data collection methods and computational data analysis methods.
Lernziele/Lernergebnisse	<p>Acquisition of the following knowledge and skills:</p> <ul style="list-style-type: none"> <li>• Ability to critical reflect on a selected research problem from applied computational social sciences on the basis of appropriate references, especially scientific original articles and the applied methodology.</li> <li>• Ability to create a comprehensive written report on a given special topic from applied computational social sciences with clarity and reasonable formalisms, within a given time frame and of a defined length.</li> <li>• Ability to participate actively in discussions on special topics of applied computational social sciences in courses with compulsory attendance.</li> <li>• Ability plan and carry out a vivid oral presentation of a specialized topic from applied computational social sciences using appropriate media and examples within a given time frame.</li> </ul>
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Knowledge comparable to the modules "Statistics and Data Science Methods" and "Social Data Science"
Literatur	-
Sprache	Englisch
Prüfungsbedingungen	Written homework and presentation (100 %). Active participation in the course is a requirement to take part in the exam (e.g. protocol of session, homework assignment, essay, oral presentation...).
Sonstiges	-
Modulverantwortung	<ul style="list-style-type: none"> <li>• Modulangebotsorganisation: LeMa-Team Philosophische Fakultät, <a href="mailto:modulangebotsorganisation@fb7.rwth-aachen.de">modulangebotsorganisation@fb7.rwth-aachen.de</a></li> <li>• Modulverantwortliche: Prof. Claudia Wagner</li> </ul>
ECTS Credits	4

- Seminar aus anderen Fakultäten/Fachgruppen
- + Current Topics in Applied Computational Social Sciences (7023944)

Kontaktzeit (SWS)	2
Prüfungsdauer (min)	-
Gesamtstunden (h)	120,0
Präsenzstunden (h)	30,0
Selbststudium (h)	90,0

### ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Seminar Current Topics in Applied Computational Social Sciences (702394401)	3. Semester	keine Semesterempfehlung	-	2
Exam Current Topics in Applied Computational Social Sciences (702394402)	3. Semester	keine Semesterempfehlung	4	0

## + Deutschkurs (1215734)

Modultitel	Deutschkurs (Wahlpflichtfach)
Kennung	1215734
Version	Angelegt über RWTH API als 1
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2001
Gültig bis	-
Modulniveau	Master
Inhalt	Studierende können Grundkenntnisse der deutschen Sprache erwerben bzw. bestehende Sprachkenntnisse erweitern und je nach Anfangsniveau vertiefen. Die Unterrichtsthemen reichen von interpersoneller Kommunikation und Alltagssituationen, über die Begegnung mit Deutschland und Europa bis zu interkulturellen Unterschieden. Im Unterricht werden auch die Spezifik des deutschen Hochschulsystems und studienspezifische Inhalte abhängig vom Sprachniveau behandelt. Bei der Wortschatzarbeit wird der Fokus auf den Transfer des Vokabulars in verschiedene Kontexte und Übungsformen gerichtet. Die Erarbeitung grammatischer Strukturen ist an die Themen und Sprachhandlungen gebunden, die den kommunikativen Bedürfnissen entsprechen.
Lernziele/Lernergebnisse	Am Ende des Kurses können Studierende eine Vielfalt von niveauspezifischen kommunikativen Handlungen im akademischen und universitären Kontext ausführen. Darüber hinaus werden Studierende einen Einblick in die deutsche und europäische Kultur bekommen und passende interkulturelle Kompetenzen erwerben.
Teilnahmebedingungen (studiengangsspezifisch)	Keine.
(empfohlene) Voraussetzungen	Keine.
Literatur	Das zu erwerbende Lehrwerk wird in der ersten Unterrichtsstunde vorgestellt. Darüber hinaus werden zusätzliche Kursmaterialien von dem Lehrer im Laufe des Semesters im Unterricht oder im L2P-Lernraum zur Verfügung gestellt.
Sprache	Deutsch
Prüfungsbedingungen	Voraussetzung für einen Leistungsnachweis ist die regelmäßige Teilnahme und das Bestehen kursinterner Leistungskontrollen (einer Zwischenprüfung und einer Abschlussprüfung). Die Benotung erfolgt auf der Basis der erreichten Punkte in der Zwischenprüfung (Klausur oder mündliche Prüfung) und der Abschlussprüfung (Klausur oder mündliche Prüfung). In den Sprachkursen besteht Anwesenheitspflicht.
Sonstiges	-
Modulverantwortung	Sprachenzentrum
ECTS Credits	4
Kontaktzeit (SWS)	4
Prüfungsdauer (min)	15-45 (mündlich/oral)   90-120 (schriftlich/written)
Gesamtstunden (h)	120,0

## + Deutschkurs (1215734)

Präsenzstunden (h)	60,0
Selbststudium (h)	60,0

## ● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Prüfung Deutschkurs (121573401)	1. Semester	keine Semesterempfehlung	4	0

## ▲ Angebotsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Lecture/Exercise German Language Course	1. Semester	keine Semesterempfehlung	-	4

## + Masterarbeit (1212703)

Modultitel	Masterarbeit (Pflichtfach)
Kennung	1212703
Version	v2
Dauer (Semester)	Einsemestrig
Turnus (Semester)	Wintersemester/Sommersemester
Gültig von	Wintersemester 2024
Gültig bis	-
Modulniveau	Master
Inhalt	Die Master-Arbeit besteht aus einer schriftliche Arbeit der Kandidatin bzw. des Kandidaten. Die Ergebnisse der Master-Arbeit präsentiert die Kandidatin bzw. der Kandidat mit einem Abschlussvortrag im Rahmen eines Master-Vortragskolloquiums.
Lernziele/Lernergebnisse	Die Arbeit soll zeigen, dass die Kandidatin bzw. der Kandidat in der Lage ist, ein Problem innerhalb einer vorgegebenen Frist nach wissenschaftlichen Methoden unter Anleitung selbstständig zu bearbeiten.
Teilnahmebedingungen (studiengangsspezifisch)	-
(empfohlene) Voraussetzungen	Das Thema der Master-Arbeit kann erst ausgegeben werden, wenn 60 CP erreicht sind.
Literatur	-
Sprache	Deutsch
Prüfungsbedingungen	Die Modulprüfung besteht aus den folgenden Teilleistungen: Schriftliche Ausarbeitung der Masterarbeit (27 CP) und Kolloquium (3 CP).
Sonstiges	-
Modulverantwortung	-
ECTS Credits	30
Kontaktzeit (SWS)	0
Prüfungsdauer (min)	-
Gesamtstunden (h)	900,0
Präsenzstunden (h)	,0
Selbststudium (h)	900,0

+ Masterarbeit (1212703)

● Prüfungsknoten

Titel	Fachsemester (Studienstart Winter)	Fachsemester (Studienstart Sommer)	ECTS Credits	Kontaktzeit (SWS)
Masterarbeit (121270301)	4. Semester	keine Semesterempfehlung	30	0