

Modulhandbuch für den Studiengang

M.Sc. Informatik

Stand: 17. Dezember 2021

Inhaltsverzeichnis

1 Wahlmodule mit praktischer Ausrichtung	3
Ausgewählte Kapitel aus Datenbanken und Informationssystemen	4
Ausgewählte Kapitel in der Human-Computer Interaction	5
Betriebssysteme	7
Big Data Analytics	8
Digital Libraries und Grundlagen des Information Retrieval	9
Erfahrungsbasierte Systeme	11
Fortgeschrittene Softwaretechnik	12
Grundlagen der Computergrafik	13
Heterogeneous Computing	14
Implementierung von Information-Retrieval-Systemen	15
Implementierung von Datenbanksystemen	16
Informationsvisualisierung	18
Maschinelles Lernen	19
Modellierung und Simulation	19
Planung und Konfiguration	19
Semantische Technologien	19
Software Architectures for Enterprises	20
Spezielle Kapitel der Praktischen Informatik A	22
Spezielle Kapitel der Praktischen Informatik B	23
Spezielle Kapitel der Praktischen Informatik C	24
Spielerprogrammierung	25
Transaktionale Informationssysteme	27
Übersetzung und Analyse von Programmen	28
Verteilte Informationssysteme	29
Verteilte Künstliche Intelligenz	30
Verteilte Systeme	31
Virtual Reality und 3D Interaktion	33

INHALTSVERZEICHNIS

2 Wahlmodule mit theoretischer Ausrichtung	35
Algorithm Engineering	36
Algorithmische Geometrie	38
Approximative Algorithmen	40
Berechenbare Analysis	42
Datenkompression	43
Ereignisgesteuerte Simulation	45
Formale Sprachen A	46
Formale Sprachen B	48
Komplexitätstheorie A	50
Komplexitätstheorie B	52
Lernalgorithmen	54
Netzwerkalgorithmen	56
Vertiefende Kapitel aus Algorithmen und Datenstrukturen	57
Parameterisierte Algorithmen	59
Rechnerarithmetik	61
Spezielle Kapitel der Theoretischen Informatik A	62
Spezielle Kapitel der Theoretischen Informatik B	63
Spezielle Kapitel der Theoretischen Informatik C	64
3 Allgemeine Wahlmodule	65
Tutor-Praktikum	66
Independent Studies	67
Ergänzungsstudium 1	68
Ergänzungsstudium 2	68
4 Forschungsorientierte Wahlmodule	68
Forschungsseminar	69
Forschungsprojekt	70
Forschungspraktikum	71
5 Masterarbeit	71
Masterarbeit im MSc Informatik	72

1 Wahlmodule mit praktischer Ausrichtung

Ausgewählte Kapitel von Datenbanken und Informationssystemen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	1. - 4. Sem	unregelmäßig	1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Fachkenntnis und Methoden zu aktuellen forschungsnahen Themen im Bereich Datenbanken und Informationssysteme				
3	Inhalte Die Inhalte dieser Veranstaltung richten sich nach aktuellen Forschungsthemen der Arbeitsgruppe. Mögliche Themen sind u.a. die folgenden: <ul style="list-style-type: none"> • Organisation von und Suche in großen Archiven • Verarbeitung von semantischen Daten • Effizienzaspekte von IR-Systemen • Soziale Netze • Lernverfahren im Information Retrieval • Evaluierung von IR-Systemen • Anwendungen von DB- und IR-Methoden in der Wissenschaft • Data Science 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeit				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter Schenkel, hauptamtlich Lehrende: Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch) in Form von Standards, Büchern und Material im Internet, z.Z. keine durchgängige Empfehlung eines einzelnen Lehrbuchs.				

Ausgewählte Kapitel in der Human-Computer Interaction					
Kennnummer	Workload 150 h	Credits 5	Studien-semester beliebig	Häufigkeit des Angebots alle zwei Jahre (Wintersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 80 Studierende Ü 20 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierende werden Einblick in aktuelle Themen der Human-Computer Interaction sowie der Virtual und Augmented Reality Forschung erhalten. Im Speziellen werden u. A. folgende Themen behandelt: <ul style="list-style-type: none"> • Entwicklung und Untersuchung persuasiver, multimodaler, taktiler interaktiver Systeme • Kooperative und kollaborative computergestützte interaktive Systeme • Formale Modellierung in der Entwicklung interaktiver Systeme • Aktuelle Themen zur Forschung in 3D Interaktion, Virtual wie Augmented Reality Softskills: <ul style="list-style-type: none"> • Erwerb von Fertigkeiten und Methoden zur Bearbeitung wissenschaftlich publizierter Ergebnisse • Erwerb von Fertigkeiten und Methoden beim Durcharbeiten der Vorlesungsinhalte • Selbstständiges Arbeiten beim Lösen von Übungsaufgaben • Argumentation und Präsentation eigener Ergebnisse 				
3	Inhalte Die Vorlesung befasst sich mit ausgewählten Themen aus dem Bereich der Human-Computer Interaction sowie der Forschung im Bereich Virtual und Augmented Reality. <ul style="list-style-type: none"> • Non-standard Interaktionsmethoden und interaktive Systeme • Kooperative und kollaborative interaktive computergestützte Systeme (CSCW) • Virtual Reality • Augmented Reality • Formale Modellierung interaktiver Systeme 				
4	Lehrformen Vorlesungen, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Modul Human-Computer Interaction (optional), Modul Virtual Reality und 3D Interaktion (optional)				
6	Prüfungsformen Mündl. Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Bearbeitung der praktischen Übungen, auch in Form von Projekt und Gruppenarbeit; Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Angewandte Mathematik, Angewandte Geoinformatik, Geographie, Computerlinguistik				
9	Stellenwert der Note für die Endnote 5/180				

10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Weyers hauptamtlich Lehrende: Weyers
11	Sonstige Informationen Foliensatz mit Hinweisen auf aktuelle Literatur (meist in Englisch). Letztes Bearbeitungsdatum: 07.11.2019

Betriebssysteme					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150	5.0	1-4	Jedes Jahr	1 Semester
1	Lehrveranstaltungen Vorlesung Betriebssysteme, 2 SWS Betriebssysteme, 1 SWS		Kontaktzeit 30 15	Selbst-studium 105	geplante Gruppengröße V-60 Ü-30
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Vermittlung detaillierter Kenntnisse über moderne Systemsoftware • Einblick in die Architektur in Betriebssystemen • Fähigkeit zur Lösung anspruchsvoller Programmieraufgaben im Bereich Systemsoftware in Kleingruppen 				
3	Inhalte <ul style="list-style-type: none"> • Einführung (Rückblick wesentliche Eigenschaften moderner Systemsoftware) • Echtzeitsysteme (EDF- und RMS-Scheduling) • ManyCore-Systeme (Konsequenzen für das Scheduling) • Architektur von Betriebssystemen (Monolith vs. Mikrokernel, Software-isolierte Prozesse) • Fortgeschrittene Parallelisierungs- und Synchronisationsansätze (Parallelisierung von Schleifen, Tasks, Futures, optimierende Zwischenschichten) • Beispiele für aktuelle Architekturen • Realisierungsvarianten und Randbedingungen für einzelne OS-Kernfunktionen (virtuelle Speicherverwaltung, Thread- und Prozeßmanagement, IPC, Dateisysteme) • Exemplarische Erweiterungen und Modifikationen an existierenden Betriebssystemen mit Zugang zum Quellcode (u.a. Linux und Windows Research Kernel) • Heranführung an aktuell bearbeitete Forschungsthemen in diesem Bereich 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeiten				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Das Modul „Systemsoftware“ sollte absolviert sein.				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Sturm				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Die vorlesungsrelevante Literatur (überwiegend wissenschaftliche Artikel) wird themenbegleitend in der Vorlesung bekannt gegeben.				

Big Data Analytics					
Kennnummer	Workload 150 h	Credits 5	Studien- semester 1. - 4. Sem	Häufigkeit des Angebots jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Fachkenntnis und Methoden zur Lösung verschiedener typischer Analyseprobleme auf großen Datenmengen mit Standardsoftware, einschließlich Kenntnis der notwendigen theoretischen Grundlagen und der Funktionsweise der verwendeten Systeme sowie der typischen Algorithmen zur Datenorganisation und Auswertungsverfahren				
3	Inhalte <ul style="list-style-type: none"> • Verteilte Dateisysteme am Beispiel von HDFS • Das Map-Reduce-Programmierparadigma am Beispiel von Apache Hadoop • Umsetzung einfacherer Analyseprobleme mit Hadoop • Systeme zur Big-Data-Verarbeitung: Apache Spark, Apache Flink • Typische Analysen auf großen Datenmengen und ihre Umsetzung, z.B. Recommender Systems, Graphanalysen, Textanalysen, Machine Learning, geographische und temporale Daten • aktuelle NoSQL-Datenbanken wie Apache HBase, Apache HIVE, Apache SPARK und MongoDB 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeit				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Data and Web Mining				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Geoinformatik, Data Science				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter Schenkel, hauptamtlich Lehrende: Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in Englisch letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur in Form von Standards, Büchern und Material im Internet, z.Z. keine durchgängige Empfehlung eines einzelnen Lehrbuchs.				

Digital Libraries und Grundlagen des Information Retrieval					
Kennnummer	Workload 150 h	Credits 5	Studien-semester 1. - 4. Sem	Häufigkeit des Angebots jedes Jahr (Wintersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h		Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Metawissen über Informatik-Fachpublikationen <ul style="list-style-type: none"> o Fachgesellschaften / Verlage / Zeitschriften / Tagungen o Wissenschaftliches Publizieren & Internet - Bibliothekswesen • Grundkenntnisse über Information Retrieval (IR) und Web Mining • Faktenwissen über theoretische Grundlagen der IR-Modelle • Methodisches Wissen über <ul style="list-style-type: none"> o die Anwendung von IR-Systemen zur Textsuche o die experimentelle Evaluation von Retrieval-Mechanismen o die Anwendung von IR-Systemen in aktuellen Szenarien 				
3	Inhalte <ul style="list-style-type: none"> • Informatik-Fachpublikationen: Institutionen, typische Tagungen & Journale, Ranking, Literaturdatenbanken • Digitale Bibliotheken: Benutzerschnittstellen, Organisation, typische Dienste • Daten, Wissen, Information • Suchprobleme, Suchstrategien, Information Seeking • Vorverarbeitung von Dokumenten • Retrievalmodelle (Boolesches Modell, Vektorraummodell, probabilistische Modelle, algebraische Modelle) • Bewertung von Retrievalsystemen • Websuchmaschinen: Aufbau eines Webcrawlers, Pagerank-Ansatz, HITS-Ansatz • Metasuchmaschinen, verteilte Suchmaschinen • Recommender Systems • Suche auf multimedialen Inhalten • Soziale Netze • Text Mining und andere Algorithmen auf Texten (z.B. Sentimentanalyse, Zusammenfassung, Plagiatsuche) • Suchmaschinenoptimierung • Praktischer Einsatz von Suchmaschinentechnologie am Beispiel von Lucene 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeit				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				

8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Data Science, Digital Humanities
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Schenkel, hauptamtlich Lehrende: Schenkel, Ley
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch). Derzeitige primäre Literaturempfehlung: <ul style="list-style-type: none"> • Christopher Manning, Prabhakar Raghavan, Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

Erfahrungsbasierte Systeme

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Fortgeschrittene Softwaretechnik					
Kennnummer	Workload 300 h	Credits 10	Studien- semester 1. oder 3. Sem.	Häufigkeit des Angebots Jedes 2. Jahr (Wintersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 6 SWS / 90 h		Selbststudium 210 h	geplante Gruppengröße V 60 Studierende Ü 30 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Kenntnis aktueller Entwicklungen in der Softwaretechnik und Fähigkeit, diese zu beurteilen • Fähigkeit fortgeschrittene Techniken einzusetzen • Fähigkeit empirische Studien zu entwerfen und einzuschätzen 				
3	Inhalte <ul style="list-style-type: none"> • Bad smells, Anit-Patterns, Refactorings • Aspektorientierte Softwareentwicklung • Agile Softwareentwicklung, Extreme Programming • Methoden und Ergebnisse der empirischen Softwaretechnik • Weitere aktuelle Themen aus der Softwaretechnik 				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, eine Übungsgruppe; Praktikum: i.d.R. Erstellen eines laufigen Programmes samt Dokumentation, Durchführung von Studien oder Experimenten				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung, möglich ist auch ein Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussklausur bzw. der mündlichen Prüfung bzw. ausreichende Benotung des Portfolios, sowie Erreichen einer Mindestpunktzahl bei den Übungsaufgaben.				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach/Nebenfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 10/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 6.1.2018 Literatur: <ul style="list-style-type: none"> • Standardliteratur (z.B. Martin Fowler et al. Refactoring: Improving the Design of Existing Code. Addison-Wesley, 1999), • sowie aktuelle Forschungsartikel zu den verschiedenen Themen 				

Grundlagen der Computergrafik					
Kennnummer	Workload 150 h	Credits 5	Studien- semester 2. oder 4. Sem.	Häufigkeit des Angebots jedes Jahr (Sommersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h		Selbststudium 105 h	geplante Gruppengröße V 60 Studierende Ü 30 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Kenntnis der Grundlagen der 2D und 3D Computergra k • Fähigkeit mit Hilfe von Werkzeugen, 3D-Modelle und Computeranimationen zu erstellen 				
3	Inhalte <ul style="list-style-type: none"> • Grundlagen: 2D Rastergra k, Vektorgra k • Bild und Videoformate • Grundlagen: 3D Computergra k <ul style="list-style-type: none"> • Renderingpipeline: Beleuchtung, Transformationen, Projektionen, Texturen • Raytracing • Modellierungssprachen und -werkzeuge • Grundlagen der Computeranimation 				
4	Lehrformen Vorlesungen, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung, möglich ist auch ein Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussklausur bzw. der mündlichen Prüfung bzw. ausreichende Benotung des Portfolios, sowie Erreichen einer Mindestpunktzahl bei den Übungsaufgaben.				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach/Nebenfach/Lehramt), Wirtschaftsinformatik, Angewandte Mathematik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 6.11.2018 Literatur: <ul style="list-style-type: none"> • James D. Foley et al. Grundlagen der Computergraphik. Addison-Wesley, 1994. • Michael Bender, Manfred Brill. Computergra k. Hanser Verlag, 2006. 				

Heterogeneous Computing					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	1-4	Jedes 2. Jahr (Sommer)	1 Semester
1	Lehrveranstaltungen a) Vorlesung, 2 SWS b) Übung, 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 120 Studierende Ü 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Kenntnisse über die Vor- und Nachteile heterogener Anwendungsarchitekturen • Möglichkeiten und Grenzen hardwarenaher Programmierung • Grundlegende Programmiermodelle heterogener Anwendungen (CPU, GPU, FPGA) Softskills <ul style="list-style-type: none"> • Erwerb von Fertigkeiten und Methoden beim Durcharbeiten der Vorlesungsinhalte • Selbständiges und gruppenorientiertes Arbeiten beim Lösen von Übungsarbeiten und kleinen Projekten • Argumentations- und Präsentationsfähigkeiten in der Gruppe und in einem Seminarumfeld 				
3	Inhalte <ul style="list-style-type: none"> • Gründe für heterogene Anwendungen; Vorteile und Nachteile, Performanzaspekte, Resilience • Asymmetrische Multiprozessoren, Spezialprozessoren (DPUs, XPU, Netzwerkprozessoren) • Hardwarenahe Programmierung: Internet der Dinge (IoT), Sensornetzwerke, Ad-Hoc-Netze, Steuerungen (Roboter, Drohnen, Autonome Fahrzeuge), SoC, DSP, Echtzeitbetriebssysteme • Hardware/Software Co-Design: FPGA • GPU-Programmierung: CUDA, OpenCL • Quantencomputer • Neuromorphe Computer, moderne Analogrechner (insbesondere neuronale Netze) • Software-Defined Architectures (u.a. Netzwerke, Speicher, Radio) 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeiten.				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Portfolioprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik, Wirtschaftsinformatik, Informatik (Lehramt)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Sturm				
11	Sonstige Informationen				

Implementierung von IR-Systemen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	1. - 4. Sem	jedes Jahr (Sommersemester)	1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Methodisches / technisches Wissen über <ul style="list-style-type: none"> ▪ typische Datenstrukturen und Algorithmen für IR-Systeme ▪ Kompressionsmethoden für Text ▪ Algorithmen für approximative Textsuche ▪ Verarbeitung von Multimedia-Dokumenten 				
3	Inhalte <ul style="list-style-type: none"> • Motivation, Stringalgorithmen • Aufbau und effiziente Konstruktion invertierter Dateien • Effiziente Anfrageverarbeitung in verschiedenen IR-Modellen (TAAT, DAAT) • Top-k-Algorithmen, z.B. Fagin's Algorithmen • Verlustfreie Kompression, z.B. Huffman (Basisalgorithmus, adaptiv, kanonisch), Zip-Lempel, arithmetische Kodierung • Approximate String Matching, z.B. Edit-Distanzen, Dynamische Programmierung, agrep, längste gemeinsame Teilsequenz • Suffix-Bäume & Arrays, Verwendung für String-Matching Probleme • Dateiformate: Postscript, PDF, Bildformate (GIF, JPEG) • Multimedia: diskrete Cosinus-Transformation. JPEG/MPEG 				
4	Lehrformen Vorlesungen, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: „Digitale Bibliotheken und Grundlagen des Information Retrieval“				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Geoinformatik, Computerlinguistik, Digital Humanities				
	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Ley hauptamtlich Lehrende: Ley, Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch) in Form von Standards, Büchern und Material im Internet, z.Z. keine durchgängige Empfehlung eines einzelnen Lehrbuchs.				

Implementierung von Datenbanksystemen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	1. - 4. Sem	jedes Jahr (Wintersemester)	1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Einführung in die internen Mechanismen von Datenbanksystemen und Dateisystemen • Faktenwissen über: <ul style="list-style-type: none"> ▪ Wichtige Dateisysteme (Windows, Unix, Linux, ...) ▪ Schichtenstruktur und Algorithmen von DBMS • Methodisches Wissen über: <ul style="list-style-type: none"> ▪ Konstruktion/Tuning/Betrieb von DBMS ▪ Optimierung, Anfrageauswertung, Speicherstrukturen, Indexe 				
3	Inhalte <ul style="list-style-type: none"> • Motivation • Hardware: Festplattentechnologie • Ein Schichtenmodell für DBMS-Server • Externspeicherverwaltung & Puffersysteme • Satz- und Zugriffspfadverwaltung • Traditionelle Dateisysteme: FAT, Unix, Berkeley FFS, NTFS • Migration von Datenbanktechnologie in Dateisysteme • Journaling File Systems (ext3, NTFS, #) • Zugriffspfade: B* Bäume, Extendible Hashing, mehrdimensionale Indexe • Algorithmen zur Anfrageauswertung: Joins, Sortieralgorithmen, • Implementierung der Anfrageauswertung als Datenflussarchitektur • Algebraische und Kostenbasierte "Optimierung"/Planung der Anfrageauswertung • Standardverfahren zur Transaktionssicherheit • alternative Speicherstrategien: Columstores, In-Memory-Datenbanken 				
4	Lehrformen Vorlesungen, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Ley hauptamtlich Lehrende: Ley, Schenkel				

11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch) in Form von Standards, Büchern und Material im Internet, z.Z. keine durchgängige Empfehlung eines einzelnen Lehrbuchs.
-----------	---

Informationsvisualisierung					
Kennnummer	Workload 150 h	Credits 5	Studien-semester 2. oder 4. Sem.	Häufigkeit des Angebots jedes Jahr (Sommersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h		Selbststudium 105 h	geplante Gruppengröße V 60 Studierende Ü 30 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Kenntnis grundlegender Visualisierungstechniken • Kenntnis physiologischer und psychologischer Faktoren • Kenntnis der wichtigsten Anwendungsgebiete von Visualisierungen (Schwerpunkt: Softwarevisualisierung) 				
3	Inhalte <ul style="list-style-type: none"> • Visualisierung von textuellen und numerischen Daten • Visualisierung von hierarchischen Informationen • Visualisierung von Graphen • Visuelle Wahrnehmung • Softwarevisualisierung • Evaluation von Visualisierungstechniken 				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussklausur bzw. der mündlichen Prüfung, sowie Erreichen einer Mindestpunktzahl bei den Übungsaufgaben.				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 8.1.2013 Literatur: <ul style="list-style-type: none"> • Stephan Diehl. Software Visualization: Visualizing the structure, behaviour, and evolution of software. Springer, 2007. • Colin Ware. Information Visualization: Perception for design. Morgan Kaufmann, 2000. • Matthew O. Ward, Georges G. Grinstein, Daniel A. Keim: Interactive Data Visualization - Foundations, Techniques, and Applications. A K Peters, 2010. 				

Maschinelles Lernen

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Modellierung und Simulation

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Planung und Konfiguration

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Semantische Technologien

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Software Architectures for Enterprises (SA4E)

Kennnummer	Workload	Credits	Studien- semester	Häufigkeit des Angebots	Dauer
	150 h	5	1-4	Jedes 2. Jahr (Winter)	1 Semester
1	Lehrveranstaltungen a) Vorlesung, 2 SWS b) Übung, 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 120 Studierende Ü 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Grundlegende Techniken des Software Reuse • Einführung in relevante Kommunikationstechniken komplexer verteilter Anwendungen • Kenntnisse über "Enterprise Integration Pattern" (EIP) und Integrationstechniken (insbesondere Apache Camel) • Grundlagen über relevante verteilte Anwendungsarchitekturen Softskills <ul style="list-style-type: none"> • Erwerb von Fertigkeiten und Methoden beim Durcharbeiten der Vorlesungsinhalte • Selbständiges und gruppenorientiertes Arbeiten beim Lösen von Übungsarbeiten und kleinen Projekten • Argumentations- und Präsentationsfähigkeiten in der Gruppe und in einem Seminarumfeld 				
3	Inhalte <ul style="list-style-type: none"> • Software Reuse, Frameworks, Komponenten, Produktfamilien, Design Pattern, Process Improvement • Historisches: UDP/TCP, RPC, OO-RPC (Java RMI, .NET Remoting, COM+, Corba) • Aktuelle Kommunikationsansätze: XML-RPC (Web Services), RESTful, HATEOAS • Managed Environments (Java, .NET), .NET WCF, JavaBeans, EJBs, Spring, Spring Boot • Softwaretechniken, Reflection, RTTI, Closures, Lambda, Annotations, Dependency Injection, Inversion of Control, Asynchrone Programmierung, Domain Specific Languages (DSL), Moderne Compiler-Techniken • Integrationswerkzeuge (Apache Thrift, Apache Camel, Finacle, gRPC) • Enterprise Integration Pattern (EIP); Umsetzung mittels Apache Camel • Software Architecting, Architekturprinzipien (u.a. Kapselung, Separation, Loose Kopplung, Transparenz, Virtualisierung) Anwendungsarchitekturen (u.a. Hierarchie, n-Tier, Lambda, Kappa, Streaming), Vertikale und Horizontale Skalierung 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeiten.				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Portfolioprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik, Wirtschaftsinformatik, Informatik (Lehramt)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Sturm				
11	Sonstige Informationen				

--	--

Spezielle Kapitel der Praktischen Informatik A

Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häu gkeit des Angebots unregelmäßig	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 60 h	Selbststudium 105 h	geplante Gruppengröße V 30 Studierende Ü 1 Gruppe/30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Sto der Praktischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. • Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Praktischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). • Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums • liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Praktischen Informatik • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und Werkzeugen und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Es werden wechselnde Vertiefungen im Bereich der Praktischen Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, eine Übungsgruppe				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl, Prof. Sturm, Prof. Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Spezielle Kapitel der Praktischen Informatik B

Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	beliebig	unregelmäßig	1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 60 h	Selbststudium 105 h	geplante Gruppengröße V 30 Studierende Ü 1 Gruppe/30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Stoff der Praktischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Praktischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Praktischen Informatik Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und Werkzeugen und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Es werden wechselnde Vertiefungen im Bereich der Praktischen Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, eine Übungsgruppe				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl, Prof. Sturm, Prof. Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Spezielle Kapitel der Praktischen Informatik C

Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	beliebig	unregelmäßig	1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 60 h	Selbststudium 105 h	geplante Gruppengröße V 30 Studierende Ü 1 Gruppe/30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Stoff der Praktischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. • Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Praktischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). • Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums • liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Praktischen Informatik • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und Werkzeugen und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Es werden wechselnde Vertiefungen im Bereich der Praktischen Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, eine Übungsgruppe				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl, Prof. Sturm, Prof. Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Spielerprogrammierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150	5.0	1-4	Jedes 2. Jahr	1 Semester
1	Lehrveranstaltungen		Kontaktzeit	Selbst-studium	geplante Gruppen-größe
	Vorlesung, 2 SWS Übung, 1 SWS		30h 15h	105	V-60 Ü-20
2	Lernergebnisse (learning outcomes) / Kompetenzen				
	<ul style="list-style-type: none"> • Überblick über die Anforderungen an moderne multimediale Spiele • Detaillierte Kenntnisse über Techniken der 3D-Graphikprogrammierung in Echtzeit • Grundlegende Programmierkenntnisse in 3D-Echtzeitgraphik (DirectX und OpenGL) • Lösung von Programmierprojekten in Kleingruppen 				
3	Inhalte				
	<p>Einführung</p> <ul style="list-style-type: none"> • Anforderungen moderne Computerspiele insbesondere im Bereich 3D-Echtzeitgraphik <p>3D-Graphik</p> <ul style="list-style-type: none"> • Mathematische Grundlagen (Vektorräume, Homogene Koordinatensysteme, Affine Transformationen) • Raytracing- und Radiosity-Verfahren • Fixed-Function und Programmable Rendering Pipeline moderner Graphikkarten • Einführung in DirectX- und OpenGL-Programmierung • Texturen und Licht • Pixel- und Vertex-Shader • Aufbau und Leistungsmerkmale moderner 3D-Hardware • Organisation komplexer Szenen (Oct-Trees, BSPs) • 3D-Engines <p>Spielphysik</p> <ul style="list-style-type: none"> • Grundlegende physikalische Gesetze (Kraft, Geschwindigkeit, Beschleunigung, Feder, Dämpfung, Reibung, Kollisionen...) • Numerische Integration und Simulation • Physikprozessoren <p>Character Animation</p> <ul style="list-style-type: none"> • Skeletthierarchie, Kinematik, Inverse Kinematik • Transformationen (Eulerwinkel, Quaternions, Lerps) • Motion Extraction • Motion Capturing • Wechselwirkungen Animation und Rendering <p>Multiplayer</p> <ul style="list-style-type: none"> • Netzwerke, Client/Server- und P2P-Ansätze • Verteilte Simulation • Dead Reckoning 				

	<ul style="list-style-type: none"> • Technische, gesellschaftliche und soziale Auswirkungen (insbesondere von Online-Spielen) <p>Spiele-KI</p> <ul style="list-style-type: none"> • Non-Player Characters • Machine Learning • Expertensysteme, CBR, Bayesian Networks • Neuronale Netze • Sensing, Thinking, Acting <p>Spiele-Entwicklung</p> <ul style="list-style-type: none"> • Übersicht und Diskussion über die einzelnen Phasen der Spielentwicklung • Wechselwirkungen Software-Engineering und Computerspiele • Wirtschaftsfaktor Computerspiele
4	Lehrformen V/S-STU/Ü
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine
6	Prüfungsformen Portfolio
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen des Portfolio
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Sturm
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch. Die vorlesungsrelevante Literatur (überwiegend wissenschaftliche Artikel) wird themenbegleitend in der Vorlesung bekannt gegeben.

Transaktionale Informationssysteme					
Kennnummer	Workload 150 h	Credits 5	Studien-semester 1. - 4. Sem	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Theoretische Grundlagen, Fachkenntnis und Methoden zur Realisierung von transaktionalen Informationssystemen, insbesondere mit Datenbanken				
3	Inhalte <ul style="list-style-type: none"> • Transaktionskonzepte • Seiten- und Objektmodell • Isolation im Seitenmodell • Concurrency-Control-Verfahren für das Seitenmodell • Isolation im Objektmodell • Concurrency-Control-Verfahren im Objektmodell • Korrektheitskriterien für Recovery-Verfahren im Seiten- und Objektmodell • Mehrphasige Recovery-Verfahren 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeit				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Implementierung von Datenbanksystemen				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Geoinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter Schenkel, hauptamtlich Lehrende: Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch) in Form von Standards, Büchern und Material im Internet Primäres Lehrbuch: Gerhard Weikum, Gottfried Vossen: Transactional Information Systems, Morgan Kaufman, 2001				

Übersetzung und Analyse von Programmen					
Kennnummer	Workload 300 h	Credits 10	Studien- semester beliebig	Häufigkeit des Angebots jedes 2. Jahr (Wintersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 6 SWS / 90 h		Selbststudium 210 h	geplante Gruppengröße V 60 Studierende Ü 30 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Kenntnis der Struktur von Compilern • Kenntnis der gängigen Erzeugungsverfahren für die verschiedenen Phasen eines Compilers • Verständnis der Übersetzung einer realen Programmiersprache (z.B. Java) • Kenntnis verschiedener Analyseverfahren 				
3	Inhalte <ul style="list-style-type: none"> • Struktur eines Compilers • Lexikalische und syntaktische Analyse • Datenflussanalyse • Übersetzung von Java • Abstrakte Maschinen, insbesondere die JVM • Erkennen von Code-Klonen • Erkennen von Entwurfsmustern 				
4	Lehrformen Vorlesungen, alternativ Lesezirkel (Reading Group), Selbststudium von ergänzender Literatur, eine Übungsgruppe; Praktikum: i.d.R. Erstellen eines lauffähigen Programmes samt Dokumentation				
5	Teilnahmevoraussetzungen keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung, möglich ist auch ein Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussklausur bzw. der mündlichen Prüfung bzw. ausreichende Benotung des Portfolios, sowie Erreichen einer Mindestpunktzahl bei den Übungsaufgaben.				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach/Nebenfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 10/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Prof. Diehl				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 6.11.2018 Literatur: <ul style="list-style-type: none"> • Reinhard Wilhelm, Dieter Maurer. Übersetzerbau. Springer-Verlag. 1992 • Stephan Diehl. A Formal Introduction to the Compilation of Java. Software, Practice & Experience, Wiley, 1998. • sowie aktuelle Forschungsartikel zu Themen wie Erkennung von Codeklonen oder Entwurfsmustern. 				

Verteilte Informationssysteme					
Kennnummer	Workload 150 h	Credits 5	Studien-semester 1. - 4. Sem	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße V 300 Studierende Ü 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Fachkenntnis und Methoden zur Realisierung von verteilten Informationssystemen, insbesondere verteilten relationalen Datenbanken, in verschiedenen Situationen				
3	Inhalte <ul style="list-style-type: none"> • verteilte relationale Datenbanken - Fragmentierung, Allokation, Anfrageausführung • Replikationsverfahren • verteilte Transaktionen, Zweiphasen-Commit-Protokoll, Serialisierbarkeit • Cloud Computing, Cloud-Datenbanken, Multi-Tenant-Systeme • Informationsintegration • föderierte Informationssysteme • Verarbeitung von Datenströmen • Verteilte IR-Systeme • Crowdsourcing-Methoden, Social Games 				
4	Lehrformen Vorlesungen, Übungen, Projekt- und Gruppenarbeit				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Implementierung von Datenbanksystemen, Verteilte Systeme				
6	Prüfungsformen Abschlussklausur oder Mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erreichen einer Mindestpunktzahl bei den Übungsaufgaben sowie Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Geoinformatik, Data Science				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter Schenkel, hauptamtlich Lehrende: Schenkel				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch letztes Bearbeitungsdatum: 23.08.2017 Foliensatz mit zahlreichen Hinweisen auf aktuelle Literatur (meist in Englisch) in Form von Standards, Büchern und Material im Internet, z.Z. keine durchgängige Empfehlung eines einzelnen Lehrbuchs.				

Verteilte Künstliche Intelligenz

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

Verteilte Systeme					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150	5.0	1-4	Jedes Jahr	1 Semester
1	Lehrveranstaltungen Vorlesung, 2 SWS Übung, 1 SWS		Kontaktzeit 30h 15h	Selbst-studium 105h	geplante Gruppengröße V-60 Ü-25
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Vermittlung detaillierter Kenntnisse über die Möglichkeiten und Grenzen verteilter Systeme • Einblick in gängige Middleware-Architekturen • Fähigkeit zur Lösung anspruchsvoller Aufgabenstellungen aus dem Bereich verteilter Systeme in Kleingruppen. 				
3	Inhalte <ul style="list-style-type: none"> • Motivation • Grenzen verteilter Systeme (fehlender globaler Zustand, fehlende gemeinsame Zeit)Vorteile verteilter Systeme (Leistungssteigerung, Fehlertoleranz, Skalierbarkeit) • Grundlagen Rechnernetze (Netzarchitekturen, Protokolle, Protokollfamilien) • Verteilte Algorithmen • Logische Uhren, Schnappschüsse, verteilter wechselseitiger Ausschluß, ... • Load Balancing (inkl. Initial Placement und Migration) • Aktive und passive Fehlertoleranz • Verteilte Dateisysteme • Cluster- und Cloud-Systeme • Systemstrukturen (u.a. zentralisierte Ansätze, Client/Server, P2P, Ad-Hoc-Netze, SOA) • Middleware-Plattformen für verteilte Anwendungen • Heranführung an aktuell bearbeitete Forschungsthemen in diesem Bereich 				
4	Lehrformen V/S-STU/Ü				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach//Lehramt), Wirtschaftsinformatik, Angewandte Mathematik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Sturm				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch.				

Literatur:

- G. Coulouris, J. Dollimore, T. Kindberg, Distributed Systems – Concepts and Design, Addison-Wesley, 2012
- A. Tanenbaum, M. van Steen, Distributed Systems: Principles and Paradigms, Prentice Hall, 2006

Virtuelle Realität und 3D Interaktion					
Kennnummer	Workload 150 h	Credits 5	Studien- semester 2. oder 3. Sem	Häufigkeit des Angebots jedes Jahr (Sommersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung b) Übung	Kontaktzeit 3 SWS / 45 h		Selbststudium 105 h	geplante Gruppengröße V 80 Studierende Ü 20 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Grundlegende Kenntnisse über Virtuelle Realität und 3D User Interfaces • Fakten- und Methodenwissen über die verwendete Hardware und Software für VR und 3D Interaktion, Interaktionskonzepte für die Interaktion mit räumlichen virtuellen Umgebungen (Selektion, Manipulation, Travel, System Control) • Praktischer Umgang mit der Entwicklung und Implementierung virtueller Umgebungen Softskills: <ul style="list-style-type: none"> • Erwerb von Fertigkeiten und Methoden beim Durcharbeiten der Vorlesungsinhalte • Selbstständiges Arbeiten beim Lösen von Übungsaufgaben • Argumentation und Präsentation eigener Ergebnisse 				
3	Inhalte <ul style="list-style-type: none"> • Motivation und Begriffsbildung • 3D Welten und Computergraphik • 3D Sehen und Wahrnehmung • Zentrale Konzepte der Virtual Reality • Display Hardware • Tracking Hardware und Systeme • Interaktions- und Eingabehardware • 3D Benutzerschnittstellen und Interaktion <ul style="list-style-type: none"> o Selektion und Manipulation o Navigation o System Control • Empirische Evaluation von VR Anwendungen 				
4	Lehrformen Vorlesungen, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Modul Computergraphik (optional)				
6	Prüfungsformen Mündl. Prüfung, Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Bearbeitung der praktischen Übungen, auch in Form von Projekt und Gruppenarbeit; Bestehen der Abschlussprüfung				
8	Verwendung des Moduls Informatik (Kernfach/Lehramt), Wirtschaftsinformatik, Angewandte Mathematik, Angewandte Geoinformatik, Geographie, Computerlinguistik				
9	Stellenwert der Note für die Endnote 5/180				
10	Modulbeauftragte/r und hauptamtlich Lehrende				

	Modulbeauftragter: Weyers hauptamtlich Lehrende: Weyers
11	<p>Sonstige Informationen</p> <p>Foliensatz mit Hinweisen auf aktuelle Literatur (meist in Englisch). Primäre Empfehlung z.Zt.:</p> <ul style="list-style-type: none"> • The VR Book, Jerald • Virtual und Augmented Reality, Dörner, Broll, Grimm, Jung • 3D User Interfaces – Theory and Practice, LaViola, Bowmann, Kruijff, Poupyrev, McMahan • Experimental Design, Cunningham, Wallraven <p>Letztes Bearbeitungsdatum: 29.01.2019</p>

2 Wahlmodule mit theoretischer Ausrichtung

Algorithm Engineering					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 60 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Das Gebiet der Algorithmen und Datenstrukturen wird traditionell der theoretischen Informatik zugeordnet. Die hier entworfenen und untersuchten Algorithmen verwenden häufig komplizierte Datenstrukturen und haben selten einen direkten Bezug zur Praxis, wo die asymptotische Komplexität der Probleme und Algorithmen und das worst-case Verhalten selten eine Rolle spielt. Hier sind häufig andere Aspekte, wie z.B. die Effiziente Nutzung des Caches oder die leichte Implementier- und Wartbarkeit viel wichtiger. • Algorithm Engineering versucht die Kluft zwischen Theorie und Praxis zu schließen. Entsprechend werden wir in dieser Vorlesung Algorithmen und Datenstrukturen betrachten, die in der Praxis tatsächlich Verwendung finden. 				
3	Inhalte <ul style="list-style-type: none"> • Algorithm Engineering, was ist das ? • Implementierung grundlegender Datenstrukturen • Cache-Effizienz • Effizientes Sortieren • Mengen, Dictionaries und Priority Queues <ul style="list-style-type: none"> o Graphen und Netzwerke o Repräsentation von Graphen und Netzwerken o grundlegende Graphalgorithmen o Kürzeste Wege o Flussprobleme • Algorithmische Geometrie • Exaktes geometrisches Rechnen • Programmier Techniken zur effizienten Implementierung von Algorithmen 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Näher				

11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Literatur: <ul style="list-style-type: none">• Cormen, Leiserson, Rivest: <i>Introduction to Algorithms</i>, Addison-Wesley, 1986• Mehlhorn, Näher: <i>LEDA, a platform for combinatorial and geometric computing</i>, Cambridge University Press, 1999
-----------	--

Algorithmische Geometrie					
Kennnummer	Workload 300 h	Credits 10	Studien-semester beliebig	Häufigkeit des Angebots 2-jährlich (Wintersemester)	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 60h / 4 SWS b) Übung 30h / 2 SWS		Kontaktzeit 6 SWS / 90 h	Selbststudium 210 h	geplante Gruppengröße 60 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Erlernen verschiedener Algorithmen und Datenstrukturen für geometrische Probleme, sowie deren Entwurf, Analyse und Anwendung. • Einschätzen der Besonderheiten diskreter geometrischer Probleme und Lösungen, etwa im Vergleich zu numerischen Verfahren, • Entwurf und Implementierung neuer Verfahren für bestimmte Anwendungen • Einsatz des Repertoires der entwickelten Datenstrukturen und Methoden für neue Probleme 				
3	Inhalte <p>Die Vorlesung behandelt den Entwurf, die Analyse und die Implementierung von Algorithmen und Datenstrukturen für geometrische Probleme. Dabei werden grundlegende Vorgehensweisen und Paradigmen, wie „Teile und Beherrsche“, Plane-Sweep, Dualität, und Randomisierung vorgestellt und auf Problemstellungen aus verschiedenen Bereichen der graphischen Datenverarbeitung angewandt, wie z.B. die Berechnung konvexer Hüllen, Bewegungsplanung für Roboter, Eliminierung von verborgenen Linien und Flächen, Boolesche Operationen auf Polygonen oder die Berechnung der nächsten Nachbarn.</p> <p>Ein zentrales Problem bei der Implementierung von geometrischen Algorithmen ist die Tatsache, dass Computer keine beliebig genauen reellen Zahlen sondern nur Fließkommazahlen zur Verfügung stellen. Die dadurch entstehenden Rundungsfehler können nicht nur zu ungenauen Ergebnissen sondern zum völligen Versagen der Programme führen. Dieses Robustheitsproblem wird in der Vorlesung genauer untersucht und es werden Methoden zu seiner Lösung entwickelt.</p>				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Algorithmen und Datenstrukturen				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 10/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Näher				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Literatur:

- Cormen, Leiserson, Rivest: *Introduction to Algorithms*, Addison-Wesley, 1986
- Klein: *Algorithmische Geometrie*, Oldenburg, 1997
- Mehlhorn, Näher: *LEDA, a platform for combinatorial and geometric computing*, Cambridge University Press, 1999
- O'Rourke: *Geometry in C*, Cambridge University Press, 1994

Approximative Algorithmen					
Kennummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. praktischer Umgang mit dem grundlegenden P/NP-Problem der (Theoretischen) Informatik 2. Neben der Vermittlung von Faktenwissen wird Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 3. Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums 4. liefert Grundlagen für erfolgreiche Masterarbeiten am Lehrstuhl 5. Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Es werden verschiedene Techniken zum Entwurf und zur Analyse approximativer Algorithmen vorgestellt. Zum näheren Ablauf: <ul style="list-style-type: none"> ➤ Motivation / Einführung: <ul style="list-style-type: none"> • P/NP Problematik • Optimierungsprobleme • Das Knotenüberdeckungsproblem • lokale Verhältnisse (local ratio) ➤ Grundlagen: <ul style="list-style-type: none"> • Fragestellungen / Aufgaben bei Optimierungsproblemen • Reduktionen ➤ Grundtechniken / Algorithmenklassen: <ul style="list-style-type: none"> • Greedy-Algorithmen • Aufteilen und Anordnen (Partitionieren / Scheduling) • Lokale Suche • Lineares Programmieren • Dynamisches Programmieren ➤ Approximationstheorie / Approximationsklassen <ul style="list-style-type: none"> • absolute / relative Approximation (APX) • Gap-Technik (Grenzen der Approximierbarkeit) • Approximationsschemata: PTAS und FPTAS • verschiedene Reduktionsbegriffe Abschließend wird in einem Kapitel mit wechselnden Inhalten knapp in einen Bereich der aktuellen Forschung auf dem Gebiet approximativer Algorithmen eingeführt.				

4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine
6	Prüfungsformen mündliche Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Die Vorlesung orientiert sich an den Kapiteln 1-5+8 des Lehrbuchs / Kompendiums: G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi: Complexity and Approximation; Springer 2003

Berechenbare Analysis					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Methodisches Wissen <ul style="list-style-type: none"> Berechenbarkeit auf nicht-abzählbaren Mengen Transferkompetenz <ul style="list-style-type: none"> Kombination der Gebiete Berechenbarkeit (Informatik) und Analysis (Mathematik) 				
3	Inhalte Viele mathematische Modelle in Wissenschaft und Technik verwenden die reellen Zahlen und darauf aufbauende Strukturen. Die berechenbare Analysis beschreibt und untersucht, wie man auf solchen Strukturen mit digitalen Computern rechnen kann. Es sollen u.a. folgende Themen behandelt werden: <ul style="list-style-type: none"> Typ-2-Berechenbarkeit, berechenbare reelle Zahlen und Funktionen, Darstellungen überabzählbarer Mengen, Zusammenhänge zwischen Berechenbarkeit und Stetigkeit, Komplexität reeller Zahlen und Funktionen, Differentiation und Integration, andere Berechenbarkeitsmodelle für die Analysis 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach), Angewandte Mathematik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Norbert Müller				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 12.2.2020 Literatur: Originalveröffentlichungen sowie insbesondere die folgenden Bücher und Artikel: <ul style="list-style-type: none"> Vasco Brattka, Peter Hertling, and Klaus Weihrauch. A tutorial on computable analysis. In S. Barry Cooper, Benedikt Löwe, and Andrea Sorbi, editors, <i>New Computational Paradigms: Changing Conceptions of What is Computable</i>, pages 425–491. Springer, New York, 2008. Klaus Weihrauch. <i>Computable Analysis</i>. Springer, Berlin, 2000. Ker-I Ko. <i>Complexity Theory of Real Functions</i>. Progress in Theoretical Computer Science. Birkhäuser, Boston, 1991. 				

Datenkompression					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. Umgang mit dem mathematischen Hintergrund moderner Datenkompressionsverfahren üben und deren Notwendigkeit einsehen. 2. Neben der Vermittlung von Faktenwissen wird Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 3. Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte <p>Es werden verschiedene Techniken zum Entwurf von Datenkompressionsalgorithmen vorgestellt. Zum näheren Ablauf:</p> <p><u>Hinweis:</u> Aufgrund der Dynamik des Gebietes werden wir uns erlauben, von Zeit zu Zeit einige der unten bezeichneten konkreten Kompressionsverfahren und -methoden durch modernere zu ersetzen.</p> <ul style="list-style-type: none"> ➤ Motivation / Grundlagen: <ul style="list-style-type: none"> • informationstheoretische Grundlagen (Wahrscheinlichkeitsmodell; Entropie) • Verzerrungsmaße für verlustbehaftete Kompression ➤ Präfixcodes: <ul style="list-style-type: none"> • Der Satz von McMillan/Kraft • Shannon / Shannon-Fano / Huffmancodierungen • erweiterte und adaptive Codierungen • arithmetische Codierung / PPM (prediction by partial match) ➤ Wörterbuchverfahren: <ul style="list-style-type: none"> • LZ77 / LZSS (g)zip • LZ78 / LZW gif, compress ➤ Weitere verlustfreie Techniken: <ul style="list-style-type: none"> • Burrows-Wheeler-Transformation (BWT) / Blocksortierung • bedingte Entropie: eine Anwendung bei der Fax-Codierung • spezielle Anwendungen (fortlaufende Bildübertragung; Gensequenzen; etc.) ➤ Allgemeine Überlegungen zur verlustbehafteten Codierung ➤ Skalarquantisierung <ul style="list-style-type: none"> • Gleichquantisierer • adaptive Quantisierer • Lloyd-Max Verfahren ➤ Vektorquantisierung <ul style="list-style-type: none"> • Lindo-Buzo-Gray-Verfahren • Codebuchentwurf 				

	<ul style="list-style-type: none"> • Codebuchübertragung: lohnt sich der Aufwand? ➤ Differentialcodierung und Teilbandcodierung <ul style="list-style-type: none"> • Prädikative Differentialcodierung • (Auto-)Korrelation • DPCM (auch mit Adaption) • Delta-Modulierung • Frequenzfilter • Shapiros Nullbaumübertragung • Einführung in Wavelets ➤ Transformationscodierung <ul style="list-style-type: none"> • Passfilter • Diskrete Fourier-Transformation DFT • Diskrete Cosinus-Transformation DCT • Fraktale Codierung ➤ Anwendungen / Standards: <ul style="list-style-type: none"> • JPEG • MPEG • Audiodaten
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine
6	Prüfungsformen mündliche Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Literatur: K. Sayood: Introduction to Data Compression. Morgan Kaufmann Das Buch wird laufend neu aufgelegt und ist als Standardlehrwerk etabliert. Die Vorlesungsfolien werden den Studierenden bereitgestellt.

Ereignisgesteuerte Simulation					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	150 h	5	beliebig	unregelmäßig	1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 15 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen Faktenwissen: <ul style="list-style-type: none"> • Basisalgorithmen für Simulationen, insb. zur Auswertung • Aufbau von Simulationssystemen Methodisches Wissen: <ul style="list-style-type: none"> • Verwendung von Simulationsbibliotheken • Implementierung von Verteilungen • Aufbau von Simulationsexperimenten Transferkompetenz: <ul style="list-style-type: none"> • Anwendungsgebiete für Simulationen 				
3	Inhalte Die Vorlesung bietet eine anwendungsorientierte Einführung in die ereignisgesteuerte Simulation. Ziel ist es, über die methodischen Grundlagen hinweg praxisbezogene Techniken zur Implementierung und Analyse komplexer Simulationsmodelle zu vermitteln. Inhalte sind: <ul style="list-style-type: none"> • Modellbildung • Simulationskonzepte • Verteilungen • Planung und Auswertung von Simulationen • Prozeß- und objektorientierte Simulation • Rare-Event-Simulation • Modellvalidierung • Pseudo-Zufallszahlen • Einführung in Markov-Ketten und Warteschlangentheorie 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Norbert Müller				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 12.2.2020 Literatur: Originalveröffentlichungen sowie insbesondere die folgenden Bücher und Artikel: <ul style="list-style-type: none"> • Bratley, Paul & Fox, Bennett L.& Schrage, Linus E., A Guide to Simulation (Springer, New York, 1987) • Page, Bernd, Diskrete Simulation (Springer, Berlin, 1991) • Jerry Banks, John Carson, Barry Nelson and David Nicol. <i>Discrete-event system simulation</i>. Pearson. 2005 • James J. Nutaro. <i>Building software for simulation: theory and algorithms, with applications in C++</i>. Wiley. 2010 				

Formale Sprachen A					
Kennnummer	Workload 150 h	Credits 5	Studien-semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen Stoff der Formalen Sprachen angeboten und entsprechendes Faktenwissen vermittelt. 2. Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 3. Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums 4. liefert Grundlagen für erfolgreiche Masterarbeiten am Lehrstuhl 5. Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	<p>Es werden wechselnde Vertiefungen im Bereich Formale Sprachen vorgestellt, aufbauend auf entsprechenden Veranstaltungen aus dem Bachelorprogramm.</p> <p>Mögliche Vertiefungen können sein:</p> <ul style="list-style-type: none"> ➤ Parallele Grammatiken ➤ weitere Automatenmodelle ➤ Regulierte Ersetzung ➤ Theoretische Modelle des Biocomputing ➤ Nicht-Standardmodelle für Berechenbarkeit ➤ Abstrakte Familien von Sprachen ➤ Algorithmische Fragestellungen bei Formalen Sprachen ➤ Beschreibungskomplexität von Formalen Sprachen 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				

9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Angesichts der erwünschten inhaltlichen Variabilität dieses Moduls lässt sich Literatur nicht allgemein angeben. Bei einer Fokussierung auf die Regulierte Ersetzung beispielsweise würde als Lehrbuch: J. Dassow und Gh. Paun: Regulated Rewriting in Formal Language Theory, Springer zum Einsatz kommen. Die Vorlesungsfolien werden den Studierenden bereitgestellt.

Formale Sprachen B					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen Stoff der Formalen Sprachen angeboten und entsprechendes Faktenwissen vermittelt. 2. Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 3. Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums 4. liefert Grundlagen für erfolgreiche Masterarbeiten am Lehrstuhl 5. Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalt Es werden wechselnde Vertiefungen im Bereich Formale Sprachen vorgestellt, aufbauend auf entsprechenden Veranstaltungen aus dem Bachelorprogramm. Mögliche Vertiefungen können sein: <ul style="list-style-type: none"> ➤ Automaten auf Bäumen und Baumsprachen ➤ Automaten auf unendlichen Strukturen, z.B. omega-Wörtern, und entsprechende Wortmengen ➤ Schwach kontextsensitive Grammatikmodelle (wichtig für Linguistische Datenverarbeitung) ➤ Syntaktische Verfahren bei der Mustererkennung ➤ Formale Potenzreihen ➤ Syntaktische Methoden der Bildbeschreibung 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				

9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Angesichts der erwünschten inhaltlichen Variabilität dieses Moduls lässt sich Literatur nicht allgemein angeben. Bei einer Fokussierung auf Automaten auf Bäumen und Baumsprachen beispielsweise würde als Lehrbuch: Tree Automata, Techniques and Applications zum Einsatz kommen. Dieses Buch wurde von einer ganzen Reihe von Autoren erstellt und hat den großen Vorteil, online verfügbar zu sein und auch immer wieder aktualisiert bzw. korrigiert zu werden. Weitere Materialien werden den Studierenden bereitgestellt.

Komplexitätstheorie A					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen Faktenwissen <ul style="list-style-type: none"> • Struktur der wichtigsten Komplexitätsklassen Methodisches Wissen <ul style="list-style-type: none"> • Reduktionstechniken zum Vergleich der Komplexität Transferkompetenz <ul style="list-style-type: none"> • Übertragung theoretischer Komplexitätsbetrachtungen auf reale Probleme Normativ-bewertende Kompetenzen <ul style="list-style-type: none"> • Einschätzung realer Probleme bzgl. ihrer Komplexität weitere Fertigkeiten <ul style="list-style-type: none"> • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Die im Bachelorstudium erworbenen Kenntnisse zum Thema Komplexität werden in vieler Hinsicht vertieft. Konkret werden folgende Bereiche behandelt werden: <ul style="list-style-type: none"> • Berechnungsressourcen Zeit und Speicherplatz, • Komplexitätsklassen L, NL, P, NP, PSPACE, • Klassifikation konkreter Probleme, • Reduzierbarkeit und Vollständigkeit, • Harte Probleme der obigen Komplexitätsklassen, • Hierarchiesätze, • Orakel-Maschinen und relativierte Berechenbarkeit, • Wenn möglich: Abschließend wird ein aktuelles Thema aus dem Bereich der Komplexitätstheorie behandelt. 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				

8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau und Norbert Müller
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch letzte Bearbeitung: 7.2. Empfohlene Literatur: Es gibt zahlreiche Bücher zu diesem Thema. Um eines herauszugreifen: Wir werden in etwa (abgesehen von den letzten beiden Kapiteln der Vorlesung) die ersten vier Kapitel behandeln aus G. Wechsung: Vorlesungen zur Komplexitätstheorie, Teubner 2000.

Komplexitätstheorie B					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen Faktenwissen <ul style="list-style-type: none"> • Struktur wichtiger Komplexitätsklassen Methodisches Wissen <ul style="list-style-type: none"> • Reduktionstechniken zum Vergleich der Komplexität Transferkompetenz <ul style="list-style-type: none"> • Übertragung theoretischer Komplexitätsbetrachtungen auf reale Probleme Normativ-bewertende Kompetenzen <ul style="list-style-type: none"> • Einschätzung realer Probleme bzgl. ihrer Komplexität weitere Fertigkeiten <ul style="list-style-type: none"> • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Die im Bachelorstudium erworbenen Kenntnisse zum Thema Komplexität werden in vieler Hinsicht vertieft. Konkret können folgende Bereiche behandelt werden: <ul style="list-style-type: none"> • Zum Umgang mit dem P/NP-Problem: <ol style="list-style-type: none"> 1. parameterisierte Komplexitätsklassen wie $W[t]$, $W[P]$ usf. 2. Approximations-Komplexitätsklassen wie APX etc. 3. Das PCP Theorem • Schaltkreis-Komplexitätsklassen wie NC^1 • Klassifikation konkreter Probleme, • Reduzierbarkeit und Vollständigkeit, • Harte Probleme der obigen Komplexitätsklassen, • abschließend wird ein aktuelles Thema aus dem Bereich der Komplexitätstheorie behandelt. 				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten				

	Erfolgreiche Übungsteilnahme, bestandene Modulprüfung
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau und Norbert Müller
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch letzte Bearbeitung: 7.2. Empfohlene Literatur: Kapitel 6-8 aus: G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi: Complexity and Approximation; Springer 2003 sowie einige Kapitel aus der Neuauflage von: R. Downey, M. Fellows: Parameterized Complexity; wird erscheinen 2013 oder 2014 bei Springer.

Lernalgorithmen					
Kennummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. Einführung in das Maschinelle Lernen, einem Kerngebiet der Künstlichen Intelligenz, vom Standpunkt der Theoretischen Informatik 2. Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums 3. Es wird stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 4. liefert Grundlagen für erfolgreiche Masterarbeiten am Lehrstuhl <p>Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen.</p>				
3	Inhalt <p>Es werden verschiedene Techniken zum Entwurf und zur Analyse von Lernalgorithmen vorgestellt. Zum näheren Ablauf:</p> <ul style="list-style-type: none"> ➤ Motivation / Grundlagen: <ul style="list-style-type: none"> • Was ist „Lernen“ (Induktion) ? • Ein einfaches mathematisches Modell (nach Gold) ➤ Lerntheorie: <ul style="list-style-type: none"> • Schlussfolgen; der Satz von Blum und Blum • charakteristische Mengen; der Satz von Angluin • Modellvarianten • Lernen von Funktionen ➤ Zur Identifikation regulärer Sprachen: <ul style="list-style-type: none"> • Minimal- und Quotientenautomaten • Verbandsstrukturen bei Zustandsverschmelzungsalgorithmen; Automatenlernen als Suchaufgabe • testbare Sprachen • reversible Sprachen • eine Anwendung im Bereich XML / SGML ➤ Anfragelernen: <ul style="list-style-type: none"> • Vorstellung des Lernmodells • Lernen von regulären Sprachen: der L* Algorithmus von Angluin • eine Anwendung zur Roboterorientierung • Erweiterungen für nicht-reguläre Sprachen 				

	<p>➤ Hidden Markov Models HMM</p> <ul style="list-style-type: none"> • Parameterschätzung als Lernprozess • Viterbi-Algorithmus • Anwendungen, speziell in der Bioinformatik und bei der Spracherkennung <p>➤ Wahrscheinlich annähernd korrektes Lernen: PAC</p> <ul style="list-style-type: none"> • das Lernmodell • Lernen von Formeln • Stichprobenkomplexität; Vapnik/Chervonenkis Dimension • eine PAC-Variante des Lernalgorithmus L^* <p>Wir werden die Vorlesung mit einem „freien“ Kapitel abschließen, das knapp in einen aktuellen Forschungsgegenstand einführt.</p>
4	<p>Lehrformen</p> <p>Vorlesungen, Selbststudium von ergänzender Literatur, Übungen</p>
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: keine</p> <p>Inhaltlich: keine</p>
6	<p>Prüfungsformen</p> <p>mündliche Prüfung</p>
7	<p>Voraussetzungen für die Vergabe von Kreditpunkten</p> <p>Erfolgreiche Übungsteilnahme, bestandene Modulprüfung</p>
8	<p>Verwendung des Moduls (in anderen Studiengängen)</p> <p>Informatik (Kernfach)</p>
9	<p>Stellenwert der Note für die Endnote</p> <p>5/120</p>
10	<p>Modulbeauftragte/r und hauptamtlich Lehrende</p> <p>Henning Fernau</p>
11	<p>Sonstige Informationen</p> <p>Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch</p> <p>Empfohlene Literatur: Wesentliche Teile des Buches von C. de la Higuera: Grammatical Inference. Cambridge University Press, 2010, werden behandelt; ergänzend wird empfohlen: M.J. Kearns: An Introduction to Computational Learning Theory, MIT Press, 1994.</p>

Netzwerkalgorithmen					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Erlernen der wichtigsten algorithmischen Techniken zur Lösung von Netzwerkproblemen (kürzeste Wege, Flussprobleme, Matchings) • Untersuchung der Komplexität dieser Probleme • Fähigkeiten zum Entwurf, der Analyse und Implementierung von entsprechenden Algorithmen • Untersuchung der praktischen Effizienz, insbesondere die Auswirkung verschiedener Datenstrukturen zur Darstellung von Graphen 				
3	Inhalte Die Vorlesung befasst sich mit der Entwicklung, Analyse und Implementierung von Algorithmen für Graphen und Netzwerkprobleme. Dabei sollen die wichtigsten Resultate dieses Gebiets behandelt werden, wie Algorithmen zur Berechnung maximaler und billigster Flüsse. Insbesondere sollen einige der behandelten Algorithmen unter Verwendung der LEDA-Bibliothek implementiert und experimentell untersucht werden.				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Näher				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Literatur: <ul style="list-style-type: none"> • Ahuja, Magnanti, Orlin: <i>Network Flows</i>, Prentice-Hall 1993, ISBN 0-13-617549-X • Mehlhorn, S. Näher: <i>LEDA - A platform for combinatorial and geometric computing</i>, Cambridge Press, ISBN 0-521-56329-1 				

Vertiefende Kapitel aus Algorithmen und Datenstrukturen					
Kennnummer	Workload 300 h	Credits 10	Studien-semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 60h / 4 SWS b) Übung 30h / 2 SWS	Kontaktzeit 6 SWS / 90 h	Selbststudium 210 h	geplante Gruppengröße 60 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Vorlesung vermittelt den Studierenden Fähigkeiten <ul style="list-style-type: none"> • zum Lesen und Verstehen aktueller Forschungsartikel aus dem Gebiet Datenstrukturen und Algorithmen • zum Verständnis fortgeschrittener Datenstrukturen und typischer algorithmischer Paradigmen aus verschiedenen Anwendungsgebieten • zur Analyse, zum Entwurf und zur Implementierung von effizienten Algorithmen • zur Einschätzung der praktischen Einsatzmöglichkeiten von Algorithmen und Datenstrukturen 				
3	Inhalte Die Vorlesung befasst sich mit ausgewählten Themen aus dem Gebiet der Algorithmen und Datenstrukturen. Aufbauend auf den Grundkenntnissen aus Informatik II werden wir Probleme aus folgenden Bereichen behandeln: <ul style="list-style-type: none"> • Datenstrukturen für Mengen • Graph- und Netzwerkalgorithmen • Algorithmische Geometrie • Parallele Algorithmen Außer dem Entwurf und der Analyse entsprechender Algorithmen werden auch Fragen der Implementierung bzw. Programmierung behandelt.				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungsgruppen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Lehramt), Wirtschaftsinformatik				
9	Stellenwert der Note für die Endnote 10/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Näher				

11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Literatur: <ul style="list-style-type: none">• Cormen, Leiserson, Rivest: <i>Introduction to Algorithms</i>, MIT Press 1990, ISBN 0-262-03141-8• Mehlhorn, Näher: <i>LEDA - A Platform for combinatorial and geometric computing</i>, Cambridge Press, ISBN 0-521-56329-1• Sedgewick: <i>Algorithms in C++ (Java)</i>• O'Rourke: <i>Computational Geometry in C</i>• Ahuja, Magnanti, Orlin: <i>Network Flows</i>
-----------	---

Parameterisierte Algorithmen					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Wintersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS		Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende
2	Lernergebnisse (learning outcomes) / Kompetenzen <ol style="list-style-type: none"> 1. praktischer Umgang mit dem grundlegenden P/NP-Problem der (Theoretischen) Informatik 2. Neben der Vermittlung von Faktenwissen wird Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). 3. Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums 4. liefert Grundlagen für erfolgreiche Masterarbeiten am Lehrstuhl 5. Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalte Es werden verschiedene Techniken zum Entwurf und zur Analyse parameterisierter Algorithmen vorgestellt. Zum näheren Ablauf: <ul style="list-style-type: none"> ➤ Motivation / Grundlagen: <ul style="list-style-type: none"> • P/NP Problematik • Parameterisierte Probleme • Die Komplexitätsklasse FPT (fixed parameter tractable) • graphentheoretische Grundbegriffe ➤ Problemkerne: <ul style="list-style-type: none"> • Problemkerneigenschaft = FPT • Kernreduktion aus Datenreduktionsregeln • lineare Kerne • Greedy-Algorithmen • Parameterisierte Dualität ➤ Suchbäume und ihre Analyse: <ul style="list-style-type: none"> • das Knotenüberdeckungsproblem • systematische Verbesserung von Suchbaumalgorithmen • Laufzeitberechnung von Suchbaumalgorithmen ➤ Graphparameter: <ul style="list-style-type: none"> • Baumzerlegungen • exakte Algorithmen, parameterisiert nach der Baumweite • Algorithmen auf planaren Graphen ➤ Weitere Verfahren <ul style="list-style-type: none"> • parameterisiertes Zählen und Aufzählen 				

	<ul style="list-style-type: none"> • dynamisches Programmieren auf Teilmengen • iterative Kompression • Farbkodierung <p>➤ Parameterisierte Komplexitätstheorie</p> <ul style="list-style-type: none"> • parameterisierte Reduktionen • die Klassen $W[1]$ und $W[2]$ <p>In einem abschließenden Kapitel werden wir speziell auf einen aktuellen Forschungsgegenstand aus dem Bereich der Parameterisierten Algorithmen eingehen.</p>
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine
6	Prüfungsformen mündliche Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)
9	Stellenwert der Note für die Endnote 5/120
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau (MA) und Stefan Näher
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Lehrbücher: R. Niedermeier: Invitation to Fixed-Parameter Algorithms. Oxford University Press 2006 R. Downey und M. Fellows: Parameterized Complexity (Neuaufgabe in Vorbereitung) Die Vorlesungsfolien werden den Studierenden bereitgestellt.

Rechnerarithmetik					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots 2-jährlich im Sommersemester	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Faktenwissen: <ul style="list-style-type: none"> • Grundkonzepte bei Darstellungen von Zahlen im Computer • Methoden zum Entwurf mehrfach-genauer Algorithmen Methodisches Wissen: <ul style="list-style-type: none"> • Einblick in die Grundlagen numerischer Bibliotheken • Exakte Analysen arithmetischer Algorithmen Normativ-bewertende Kompetenzen: <ul style="list-style-type: none"> • Erkennen der Problematik arithmetisch ungenauer Algorithmen 				
3	Inhalte Grundlagen der Computerarithmetik und des wissenschaftlichen Rechnens: elementare Integer-Arithmetik – schnelle Multiplikation – modulare und redundante Zahl-Darstellungen – rationale Arithmetik – Fließkomma-Zahlen – IEEE 754/854 Fließkomma-Standards – Multiple-Precision-Arithmetik – Intervall-Arithmetik – Reduktionsmethoden in der Arithmetik – Komplexitätsbetrachtungen in der Arithmetik				
4	Lehrformen Vorlesungen, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach), Angewandte Mathematik				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Norbert Müller				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 12.2.2020 Literatur: Originalveröffentlichungen sowie insbesondere das folgende Buch: <ul style="list-style-type: none"> • Richard Brent and Paul Zimmermann. Modern Computer Arithmetic, Cambridge Monographs on Applied and Computational Mathematics, 2011 				

Spezielle Kapitel der Theoretischen Informatik A					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots unregelmäßig	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung bzw. Reading Group 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Stoff der Theoretischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. • Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). • Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums • liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Theoretischen Informatik • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalt Es werden wechselnde Vertiefungen im Bereich Theoretischer Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, Reading Group, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau, N.N.				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Spezielle Kapitel der Theoretischen Informatik B					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots unregelmäßig	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung bzw. Reading Group 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Stoff der Theoretischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. • Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). • Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums • liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Theoretischen Informatik • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalt Es werden wechselnde Vertiefungen im Bereich Theoretischer Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, Reading Group, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Abschlussklausur oder mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau, N.N.				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Spezielle Kapitel der Theoretischen Informatik C					
Kennnummer	Workload 150 h	Credits 5	Studien-semester beliebig	Häufigkeit des Angebots unregelmäßig	Dauer 1 Semester
1	Lehrveranstaltungen a) Vorlesung bzw. Reading Group 30h / 2 SWS b) Übung 15h / 1 SWS	Kontaktzeit 3 SWS / 45 h	Selbststudium 105 h	geplante Gruppengröße 25 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Aufbauend auf die Einführungsveranstaltung im Bachelor-Studium werden Ergänzungen aus dem klassischen und aktuellen Stoff der Theoretischen Informatik angeboten und entsprechendes Faktenwissen vermittelt. • Darüber hinaus wird auch stets Wert gelegt auf die Vermittlung und Einübung der Methoden innerhalb insbesondere der Theoretischen Informatik mit den entsprechenden Spezialisierungen (Methodenwissen). • Heranführung an den aktuellen Forschungsstand im Rahmen eines forschungsorientierten Studiums • liefert Grundlagen für erfolgreiche Masterarbeiten im Bereich der Theoretischen Informatik • Der Selbststudiumsanteil umfasst auch eigenständigen Umgang mit entsprechender Fachliteratur und dient so zur Einübung / Vorbereitung auf lebenslanges Lernen. 				
3	Inhalt Es werden wechselnde Vertiefungen im Bereich Theoretischer Informatik vorgestellt. Die Inhalte können sich wechselnden Gegebenheiten und Interessen der Lehrenden anpassen und streben enge Bezüge zur aktuellen Forschung an.				
4	Lehrformen Vorlesungen, Reading Group, Selbststudium von ergänzender Literatur, Übungen				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen mündliche Prüfung				
7	Voraussetzungen für die Vergabe von Kreditpunkten Erfolgreiche Übungsteilnahme, bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Henning Fernau, N.N.				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

3 Allgemeine Wahlmodule

Tutor-Praktikum (Master)					
Kennnummer	Workload 150 h	Credits 5	Studien-semester beliebig	Häufigkeit des Angebots jedes Semester	Dauer 1 Semester
1	Lehrveranstaltungen Praktikum	Kontaktzeit 4 SWS / 60h	Selbststudium 90 h	geplante Gruppengröße 5 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Tutoren lernen, wie Lehrveranstaltungen organisiert werden und welche methodischen Ziele dabei verfolgt werden. Sie lernen, komplexe fachliche Inhalte sowohl in einer größeren Gruppe (Übungsgruppe) als auch in individuellen Beratungsgesprächen zu vermitteln. Sie lernen in ihrer Tätigkeit, sich an das unterschiedliche Vorwissen und die unterschiedlichen intellektuellen Fähigkeiten der betreuten Studierenden anzupassen. Sie werden ermutigt, komplexe fachliche Zusammenhänge einfach, prägnant und wirkungsvoll zu vermitteln. Sie lernen das Erstellen von Lehrmaterialien insbesondere von Übungsaufgaben und Musterlösungen.				
3	Inhalte Ein Tutor unterstützt eine Lehrveranstaltung über den Zeitraum eines Semesters. Der Dozent der zu betreuenden Veranstaltung legt Aufgaben im Umfang von 150 Stunden im Wesentlichen aus folgenden Aufgabenbereichen fest: <ul style="list-style-type: none"> • Vorbesprechung / Planung der jeweiligen Lehrveranstaltung (4h) • Moderieren einer wöchentlichen Übungsgruppe (je 90 min) • Korrigieren der wöchentlichen Tests • Wöchentliche Beratungsstunden (je 90 Minuten) für die Hörer der Vorlesung • Teilnahme an der wöchentlichen Teambesprechung der Vorlesung, an der das gesamte Lehrpersonal teilnimmt (je 45 Minuten) • Beantwortung von Fragen zum Vorlesungsstoff und zum Übungsblatt auf der zugehörigen Lernplattform (60 Minuten pro Woche) • Einarbeitung in den Vorlesungsstoff (2 Stunden pro Woche) • Klausuraufsicht und Unterstützung bei der Klausurkorrektur (Zwischenklausur, Endklausur, Nachklausur, je 12 Stunden) • Erfinden neuer Übungsaufgaben (60 Minuten pro Woche) • Erstellung von Musterlösungen für Übungsaufgaben (90 Minuten) 				
4	Lehrformen Betreuung von Übungen				
5	Teilnahmevoraussetzungen Formal: Der Tutor muss die zu betreuende Lehrveranstaltung mit der Note 1,7 oder besser absolviert haben. Inhaltlich: Die Tutoren werden vom Dozenten ausgewählt, insbesondere nach didaktischem Interesse und didaktischer Befähigung				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
8	Verwendung des Moduls (in anderen Studiengängen) <i>M.Sc. Informatik (Kernfach)</i>				
9	Stellenwert der Note für die Endnote entsprechend der Regelung in der APOM der Universität Trier				
10	Modulbeauftragte/r und hauptamtlich Lehrende Norbert Müller / alle hauptamtlich Lehrenden des Faches Informatik				
11	Sonstige Informationen				

Independent Studies					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots jedes Semester	Dauer 1 Semester
1	Lehrveranstaltungen keine	Kontaktzeit 10 h		Selbststudium 140 h	geplante Gruppengröße 1
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Vertiefte informatik-relevante Kenntnisse Softskills: <ul style="list-style-type: none"> • Eigenverantwortliche, selbständige Wissensaneignung 				
3	Inhalte <ul style="list-style-type: none"> • Ziel des Moduls ist die selbständige Einarbeitung in ein Forschungsfeld sowie der Erwerb von vertiefenden Spezialkenntnissen, insbesondere als Vorbereitung auf die Masterarbeit. • Die Inhalte müssen vorher mit dem betreuenden Dozenten abgesprochen werden. 				
4	Lehrformen Selbststudium				
5	Teilnahmevoraussetzungen Keine				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten ausreichende Benotung des Portfolios				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach/Nebenfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Prof. Diehl Hauptamtlich Lehrende: die Dozenten der Informatik/Wirtschaftsinformatik				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch Letztes Bearbeitungsdatum: 24.06.2011				

Ergänzungsstudium 1

(Importmodul, siehe entsprechende FPO)

Ergänzungsstudium 2

(Importmodul, siehe entsprechende FPO)

4 Forschungsorientierte Wahlmodule

Forschungsseminar					
Kennnummer	Workload 150 h	Credits 5	Studien- semester beliebig	Häufigkeit des Angebots jedes Semester	Dauer 2 Semester
1	Lehrveranstaltungen Seminar	Kontaktzeit 2 SWS / 30 h		Selbststudium 120 h	geplante Gruppengröße bis ca. 15 Personen
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> • Heranführung an den aktuellen Forschungsstand Softskills: <ul style="list-style-type: none"> • Übung von Präsentationstechniken • Übung von Kommunikationsfähigkeiten in der Diskussion • Übung im Abfassen wissenschaftlicher Aufsätze durch Ausarbeitung • i.d.R Übung des Leseverständnisses englischsprachiger Literatur • Selbständiges Durcharbeiten eines oder mehrerer Forschungsartikel oder Buchkapitel • Heranführung an den aktuellen Forschungsstand 				
3	Inhalte <ul style="list-style-type: none"> • Die behandelten Problemstellungen orientieren sich an aktuellen Forschungsgebieten. 				
4	Lehrformen <ul style="list-style-type: none"> • Vorträge der Studierenden, Selbststudium von ergänzender Literatur schriftliche Ausarbeitung, Diskussion 				
5	Teilnahmevoraussetzungen Keine				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten ausreichende Benotung des Portfolios				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach)				
9	Stellenwert der Note für die Endnote 5/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Prof. Diehl Hauptamtlich Lehrende: die Dozenten der Spezialisierungen				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Forschungsprojekt					
Kennnummer	Workload 300 h	Credits 10	Studien- semester beliebig	Häufigkeit des Angebots jedes Semester	Dauer 2 Semester
1	Lehrveranstaltungen Praktikum	Kontaktzeit 4 SWS / 60 h		Selbststudium 240 h	geplante Gruppengröße bis ca. 15 Personen
2	Lernergebnisse (learning outcomes) / Kompetenzen <ul style="list-style-type: none"> Heranführung an den aktuellen Forschungsstand Softskills: <ul style="list-style-type: none"> i.d.R Übung des Leseverständnisses englischsprachiger Literatur Selbständiges Durcharbeiten eines oder mehrerer Forschungsartikel oder Buchkapitel Heranführung an den aktuellen Forschungsstand Arbeit in Kleingruppen (insbesondere Entwicklung einer arbeitsteiligen Vorgehensweise und der Implementierung von partiellen Erkenntnissen in den Gesamtprozess). Umsetzung ingenieurmäßiger Methoden und Techniken zur systematischen Entwicklung von Software-Systemen in der Praxis einzusetzen 				
3	Inhalte <ul style="list-style-type: none"> Die behandelten Problemstellungen orientieren sich an aktuellen Forschungsgebieten 				
4	Lehrformen <ul style="list-style-type: none"> Praktikum: i.d.R. Erstellen eines lauffertigen Programmes samt Dokumentation und Selbststudium entsprechender Literatur mit Einarbeitung in bestehende Software 				
5	Teilnahmevoraussetzungen Keine				
6	Prüfungsformen Portfolio				
7	Voraussetzungen für die Vergabe von Kreditpunkten ausreichende Benotung des Portfolios				
8	Verwendung des Moduls (in anderen Studiengängen) Informatik (Kernfach/Hauptfach)				
9	Stellenwert der Note für die Endnote 10/120				
10	Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragter: Prof. Diehl Hauptamtlich Lehrende: die Dozenten der Informatik				
11	Sonstige Informationen Veranstaltung i.d.R. in deutscher Sprache, bei Bedarf jedoch auf Englisch				

Forschungspraktikum

(Importmodul, siehe FPO M.Sc. Wirtschaftsinformatik)

5 Masterarbeit

Masterarbeit im MSc Informatik (Kernfach/Hauptfach)						
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots		Dauer
	900 h	30	beliebig	jedes Semester		1 Semester
1	Lehrveranstaltungen			Kontakt-zeit	Selbst-studium	geplante Gruppengröße
	Masterarbeit Kolloquium			15 h	885 h	i.d.R. Einzelarbeit
2	Lernergebnisse (learning outcomes) / Kompetenzen					
	<p>Die Studierenden</p> <ul style="list-style-type: none"> • besitzen die Fähigkeit, wissenschaftliche Methoden und Kenntnisse des Faches eigenständig anzuwenden und zu erweitern, • sind in der Lage, die Zusammenhänge des Faches zu überblicken, • können eine umfangreiche schriftliche Arbeit unter Einhaltung einer Zeitvorgabe zielorientiert planen • und sind in der Lage, diese Arbeit, mit Interpretation und Bewertung, in einem vorgegebenen Zeitraum zu erstellen. <p>Insbesondere wird auch die Schlüsselqualifikation der Organisationsfähigkeit gefördert.</p>					
3	Inhalte					
	In der Masterarbeit soll eine komplexe Problemstellung aus dem Gebiet der Informatik selbstständig unter Anwendung des Theorie- und Methodenwissens der Informatik bearbeitet und gemäß wissenschaftlicher Standards dokumentiert werden.					
4	Lehrformen					
	Selbststudium					
5	Teilnahmevoraussetzungen					
	keine					
6	Prüfungsformen					
	schriftliche Arbeit					
7	Voraussetzungen für die Vergabe von Kreditpunkten					
	erfolgreiche Präsentation der Masterarbeit in einem Kolloquium					
8	Verwendung des Moduls (in anderen Studiengängen)					
	Informatik (Kernfach/Hauptfach)					
9	Stellenwert der Note für die Endnote					
	30/120					
10	Modulbeauftragte/r und hauptamtlich Lehrende					
	Müller					
11	Sonstige Informationen					