

MODULHANDBUCH

Übersicht der Module

Informatik - Master (1-Fach, PO 2023)

Basic/Foundation Modules

Advanced Algorithms

Kryptographie

Mathematics for Modeling and Data Analysis

Theorie des maschinellen Lernens

Design, Implementation and Analysis of Computer Systems

Deterministic Network Calculus

Energy-Aware Computing Systems

High-Performance Computing on Clusters

High-Performance Computing on Multi- and Manycore Processors

Model Checking

Operating-System Concepts and Implementations

Real Time Networks and Systems

System Performance Evaluation

Algorithms, Complexity, Data

Geometrische Algorithmen

Information Theory

Komplexitätstheorie

Model Checking

Quantum Information and Computation

Computer Security

Blockchain Security and Privacy

Kryptographie auf hardwarebasierten Plattformen

Kryptographie auf hardwarebasierten Plattformen

Model Checking

Physical Attacks and Countermeasures

Quantum Information and Computation

Software Engineering and Programming Languages

Autonomous Vehicles and Artificial Intelligence

High-Performance Computing on Clusters

High-Performance Computing on Multi- and Manycore Processors

Artificial Intelligence

Autonomous Robotics: Action, Perception, Cognition

Autonomous Vehicles and Artificial Intelligence

Computational Neuroscience: Neural Dynamics

Computational Neuroscience: Single-Neuron Models

Computational Neuroscience: Vision and Memory

Deep Learning

Information Theory

Knowledge Graphs

Machine Learning: Evolutionary Algorithms

Machine Learning: Supervised Methods

Machine Learning: Unsupervised Methods

Statistisches Lernen und Data Mining

Theorie des maschinellen Lernens

Practical Modules

Practical Labs

Project

Seminars

Free Elective Moduls

Free Elective Moduls

Masterthesis

Masterarbeit und Kolloquium (INF)

| | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|------------------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Advanced Algorithms | | | | | |
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Advanced Algorithms | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Maike Buchin Lehrende: Prof. Maike Buchin | | | | | |
| Verwendung des Moduls - Master Informatik - Master Angewandte Informatik - Master Mathematik | | | | | |
| Lernziele (learning outcomes) - Fortgeschrittene Entwurfsmethoden für Algorithmen - Fortgeschritten Analysemethoden für Algorithmen - Kenntnis weiterer Datenstrukturen und Methoden zum Entwurf von Datenstrukturen - Anwendung der gelernten Methoden auf neue Probleme | | | | | |
| Inhalt In der Vorlesung betrachten wir fortgeschrittene Themen der Algorithmik. Nach einer kurzen Wiederholung bekannter Inhalte betrachten wir vor allem Graphalgorithmen, Approximationsalgorithmen und FPT-Algorithmen sowie exakte Algorithmen für NP-schwere Probleme. Ebenfalls betrachten wir einige neue und bekannte Datenstrukturen und deren Analyse. Die betrachteten Probleme dabei sind sowohl kombinatorisch, graphentheoretisch also auch geometrisch. | | | | | |
| Lehrformen Vorlesung (als Folien- und Tafelvortrag) und Übungen, in denen die vorgestellten Inhalte vertieft werden | | | | | |
| Prüfungsformen Mündliche oder schriftliche Modulabschlussprüfung (wird zu Semesterbeginn bekannt gegeben) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an Übungen | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Kryptographie Cryptography | | | | | |
| Modul-Nr./Code | Credits 8 CP | Workload 240 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Kryptographie (212017) | | | Kontaktzeit 90 h | Selbststudium 150 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Alexander May Lehrende: Prof. Alexander May | | | | | |
| Verwendung des Moduls B.Sc. IT-Sicherheit/Informationstechnik M.Sc. IT-Sicherheit/Netze und Systeme M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden haben ein Verständnis der wesentlichen mathematischen Methoden und Verfahren, auf denen moderne kryptographische Verfahren beruhen. Die Tiefe der Behandlung der Verfahren geht deutlich über das in den vorhergehenden Veranstaltungen vermittelte Maß hinaus. Die Teilnehmer sind zur Analyse und dem Design aktueller und zukünftiger kryptographischer Methoden befähigt. Zudem weisen sie ein Bewusstsein für Methodik und Mächtigkeit verschiedenster Angriffsszenarien auf. | | | | | |
| Inhalt Es wird eine Einführung in moderne Methoden der symmetrischen und asymmetrischen Kryptographie geboten. Dazu wird ein Angreifermodell definiert und die Sicherheit der vorgestellten Verschlüsselungs-, Hash- und Signaturverfahren unter wohldefinierten Komplexitätsmaßnahmen in diesem Angreifermodell nachgewiesen. Themenübersicht: <ul style="list-style-type: none"> • Sichere Verschlüsselung gegenüber KPA-, CPA- und CCA-Angreifern • Pseudozufallsfunktionen und -permutationen • Message Authentication Codes • Kollisionsresistente Hashfunktionen • Blockchiffren • Konstruktion von Zufallszahlengeneratoren • Diffie-Hellman Schlüsselaustausch • Trapdoor Einwegpermutationen • Public Key Verschlüsselung: RSA, ElGamal, Goldwasser-Micali, Rabin, Paillier • Einwegsignaturen • Signaturen aus kollisionsresistenten Hashfunktionen • Random-Oracle Modell | | | | | |
| Lehrformen Vorlesung und Übungen | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

8/99: M.Sc. IT-Sicherheit/Netze und Systeme [PO 22]

8/96: M.Sc. IT-Sicherheit/Netze und Systeme [PO 22]

8/97: M.Sc. Informatik

8/105: M.Sc. Angewandte Informatik

| Titel des Moduls: Mathematics for Modeling and Data Analysis | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|----------------------------------------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester (kein Angebot im SS 23!) | Dauer 1 Semester |
| Lehrveranstaltungen Mathematics for Modeling and Data Analysis (211047) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott | | | | | |
| Verwendung des Moduls B.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students <ul style="list-style-type: none"> • know the material covered in this course, see Content, • do have an intuitive understanding of the basic concepts and can work with that, • can communicate about all this in English | | | | | |
| Inhalt This course covers mathematical methods that are relevant for modeling and data analysis. Particular emphasis is put on an intuitive understanding as is required for a creative command of mathematics. The following topics are covered: <ul style="list-style-type: none"> • Functions and how to visualize them • Vector spaces • Matrices as transformations • Systems of linear differential equations • Qualitative analysis of nonlinear differential equations • Bayesian theory • Markov chains | | | | | |
| Lehrformen This course is given with the flipped/inverted classroom concept. First, the students work through online material by themselves. In the lecture time slot we then discuss the material, find connections to other topics, ask questions and try to answer them. In the tutorial time slot the newly acquired knowledge is applied to analytical exercises and thereby deepened. I encourage all students to work in teams during self-study time as well as in the tutorial. | | | | | |
| Prüfungsformen The course is concluded with a digital written exam. | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/168: B.Sc. Angewandte Informatik [PO 22] 5/170: B.Sc. Angewandte Informatik [PO 20] 5/97: M.Sc. Informatik | | | | | |

Titel des Moduls: Theorie des maschinellen Lernens

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Theorie des maschinellen Lernens (211052) | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden werden mit mathematischen Modellen für das maschinelle Lernen vertraut gemacht. Sie erwerben die Fähigkeit, Lernalgorithmen zu beurteilen und zu vergleichen anhand des Grades, in welchem diese (exakt beschriebene) Erfolgskriterien erreichen. Sie erwerben Techniken sowohl zum Design effizienter Lernalgorithmen als zum Nachweis der inhärenten Härte eines Problems. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none">• kennen Studierende die wichtigsten Lernmaschinen (wie zum Beispiel Support Vector Machines und verwandte Modelle),• verstehen Studierende den Unterschied zwischen empirischer und realer Fehlerrate und kennen Techniken zum Umgang mit dem Problem des overfitting der Daten (mit einem zu komplexen Modell),• können Studierende zwischen uniformer und nicht uniformer Lernbarkeit einer Hypothesenklasse unterscheiden und kennen die dazu passenden Theorien und Lernregeln. | | | | | |
| Inhalt Gegenstand der Vorlesung ist die statistik-basierte Theorie des maschinellen Lernens. Insbesondere wird die Methode der strukturierten Risikominimierung vermittelt sowie die ihr zugrunde liegenden statistischen Lehrsätze. Es werden sowohl Techniken zum Entwurf effizienter Lernalgorithmen besprochen als auch informations- oder berechnungstheoretische Barrieren, die bestimmte Lernprobleme als nicht effizient lösbar erscheinen lassen. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen mündliche Modulabschlussprüfung (30 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik 9/97: M.Sc. Informatik | | | | | |

Titel des Moduls: Deterministic Network Calculus

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Deterministic Network Calculus (211054) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Steffen Bondorf Lehrende: Prof. Dr. Steffen Bondorf | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. IT-Sicherheit/Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls werden die Studierenden in der Lage sein, <ul style="list-style-type: none">• komplexe, vernetzte Systeme als deterministische Warteschlangensysteme zu modellieren,• worst-case Leistungsanalysen von bestehenden Systemen bzw. Modellen durchzuführen,• die Herausforderungen bei der Leistungsdimensionierung von geplanten Systemen zu verstehen, &#8729; dabei die Wirkungsweise zentraler Mechanismen in Computernetzen anhand des Network Calculus zu erklären,• die vorgestellten Verfahren gegeneinander abzugrenzen und auf wissenschaftliche Fragestellungen anzuwenden. | | | | | |
| Inhalt Verteilte Systeme sind heutzutage allgegenwärtig, und ihre Vernetzung ist von grundlegender Bedeutung für die kontinuierliche Verbreitung und damit Verfügbarkeit von Daten. Die Bereitstellung von Daten in Echtzeit ist einer der wichtigsten nichtfunktionalen Aspekte, den sicherheitskritische Netze gewährleisten müssen. Die formale Verifizierung der Datenkommunikation im Hinblick auf die worst-case Deadlines ist grundlegend für die Zertifizierung von neu entwickelten x-by-Wire-Systemen. Diese Verifizierung erlaubt den Start von Flugzeugen, das Lenken von Autos ohne mechanische Verbindung und den Betrieb sicherheitskritischer Industrieanlagen. Daher wurden verschiedene Methoden für die worst-case Modellierung und Analyse von Echtzeitsystemen entwickelt. Eine davon ist der Deterministische Network Calculus (DNC), eine vielseitige Technik, die in verschiedenen Bereichen wie Paketvermittlung, Task Scheduling, System on Chip, softwaredefinierte Netzwerke, Netzwerke in Rechenzentren und Netzwerkvirtualisierung eingesetzt werden kann. DNC ist eine Methode zur Ableitung deterministischer Schranken für zwei der vorrangigsten Leistungsmetriken in Kommunikationssystemen: <ul style="list-style-type: none">• die Ende-zu-Ende-Verzögerung von Datenflüssen und• der Speicherplatz, den ein Server benötigt, um alle eingehenden Daten in einer Warteschlange zu puffern. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung über 30 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/97: M.Sc. Informatik

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

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

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Energy-Aware Computing Systems

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|-----------------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Energy-Aware Computing Systems | | | Kontaktzeit | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen none | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Timo Hönig Lehrende: Prof. Dr.-Ing. Timo Hönig | | | | | |
| Verwendung des Moduls - Master Informatik - Master IT-Sicherheit/Netze und Systeme - Master IT-Sicherheit/Informationstechnik - Master Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Students who have successfully attended the lecture and exercises have followed the learning objectives and have acquired the competence listed below. Students can <ul style="list-style-type: none">• understand the importance of electrical energy as an operating resource to computing systems• make trade-off decisions in regard to efficient system design (i.e., power demand vs. performance), in particular of operating systems• model the energy demand for individual synchronous and asynchronous operations• apply strategies to reduce the energy demand for software activities based on specific hardware properties (i.e., sleep states)• analyse software for critical sections that cause high energy demand | | | | | |

Inhalt

Electrical energy is the single most important operating resource for computer systems. Although the energy demand of computers is an invisible system property by itself, the impact of energy demand is omnipresent and obvious in various forms of appearance. Sudden system failures (i.e., system breakdowns) and recurrent standard system operations (i.e., power management) serve as practical examples. The lecture discusses the design of energy-aware computing systems and focuses on the following topics:

- power and energy management
- energy accounting
- energy demand analysis
- energy-aware operating-system architecture
- hardware power management (i.e., DVFS, throttling, sleep states)
- thermal management
- storage and file systems
- memory management
- network, wireless, and protocols
- energy-aware server/cluster
- compiler optimisations and code transformation
- display technology
- electricity grid

The lecture is linked to the exercises through research papers. The students read the papers in preparation for lectures. From there, the research papers are the basis for discussion and build the starting point for the assignments of the exercises. As part of the exercises, the students apply concepts and strategies from the research papers to systems and evaluate the impact on the system's energy efficiency.

Lehrformen

The lecture is held with a seminar character. Research papers on energy-aware computing and system design are prepared by the students and discussed and analysed during the sessions. Additionally, the lecture imparts theoretical knowledge of fundamental concepts for the individual topics.

As part of the exercises, the students apply their acquired knowledge by adapting system software and system configurations to improve energy efficiency. They analyse the results by conducting performance and energy-demand evaluations.

Prüfungsformen

Oral exam (30 minutes)

Voraussetzungen für die Vergabe von Credits

- exam
- successful completion of the exercises

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

Titel des Moduls: High-Performance Computing on Clusters

| | | | | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Clusters | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program and employ parallel computing clusters. Theoretical properties of distributed-memory systems and programming patterns are conveyed as well as the practical implementation. | | | | | |
| Inhalt The lecture deals with the parallelization on cluster computers. Distributed-memory programming concepts (MPI) are introduced and best-practice implementation is presented based on applications from scientific computing including the finite element method and machine learning. Special attention is paid to scalable solvers for systems of equations on distributed memory systems, focusing on iterative schemes such as simple splitting methods (Richardson, Jacobi, Gauß-Seidel, SOR), Krylov-methods (Gradient descent, CG, BiCGStab) and, in particular, the multigrid method. The mathematical foundations for iterative solvers are reviewed, suitable object-oriented interface structures are developed and an implementation of these solvers for modern parallel computer architectures is developed. Numerical experiments and self-developed software implementations are used to discuss and illustrate the theoretical results. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Written Exam (120 minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: High-Performance Computing on Multi- and Manycore Processors

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Multi- and Manycore Processors | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program multi- and manycore processors employing multi-threaded execution and handling shared-memory access patterns. Theoretical properties are conveyed as well as practical implementation. Via presentations of selected topics, students attain the ability to survey and acquire knowledge on advanced scientific topics independently and are qualified to illustrate such topics in the form of a presentation and numerical examples. | | | | | |
| Inhalt The lecture addresses parallelization for multi- and manycore processors. Thread-based programming concepts (pthreads, C++11 threads, OpenMP, OpenCL) are introduced and bestpractice implementation aspects are highlighted based on applications from scientific computing. In the first part, the lecture provides an overview on relevant data structures, solver techniques and programming patterns from scientific computing. An introduction to multithreading programming on multicore systems is then provided with special attention to shared-memory aspects. Parallelization patterns are discussed and highlighted. Numerical experiments and self-developed software implementations are used to discuss and illustrate the presented content. In the second part, students are assigned advanced topics for sharedmemory computation from the engineering science including finite element methods and artificial intelligence. Based on a scientific paper, students present their topic to the lecture audience in form of a beamer presentation and numerical illustrations. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Semesterbegleitend; Hausarbeit | | | | | |
| Voraussetzungen für die Vergabe von Credits mit mindestens ausreichend bewertete Hausarbeit | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Model Checking

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Model Checking | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Dr. Nils Vortmeier Marko Schmellenkamp | | | | | |
| Verwendung des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Mathematik | | | | | |
| Lernziele (learning outcomes) Die Studierenden lernen wie sich verteilte Systeme durch Transitionssysteme modellieren und Eigenschaften in logischen Spezifikationssprachen wie LTL und CTL spezifizieren lassen. Sie sollen elementare Algorithmen zur Überprüfung von Eigenschaften in Transitionssystemen kennenlernen. Sie sollen ein Verständnis für die Möglichkeiten und Grenzen des Model Checkings entwickeln, und in die Lage versetzt werden, sich eigenständig mit fortgeschrittenen Methoden des Model Checkings auseinanderzusetzen. | | | | | |
| Inhalt Wie kann die Korrektheit von Software und Hardware formal überprüft werden? Im Model Checking werden Software- und Hardware-Module durch Transitionssysteme formalisiert; gewünschte Eigenschaften mit Hilfe logischer Formalismen formal beschrieben; und mit Hilfe von Algorithmen automatisiert überprüft, ob ein Transitionssystem eine formal spezifizierte Eigenschaft besitzt. In dieser Veranstaltung werden die theoretischen Grundlagen des Model Checkings vermittelt, mit einem Fokus auf logik-basierten Spezifikationssprachen. Die Spezifikationssprachen LTL und CTL werden eingeführt, ihre Ausdrucksstärke untersucht, und die wichtigsten algorithmischen Ansätze für das Model Checking vorgestellt. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Abschlussprüfung; mündliche Prüfung (20-30min) oder schriftliche Klausur (120min) in Abhängigkeit der Teilnehmerzahl | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulprüfung. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/170: B.Sc. Informatik [PO 22]

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

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

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

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

Titel des Moduls: Operating-System Concepts and Implementations

| | | | | | |
|----------------------------------------------------------------------------|------------------------|--------------------------|-----------------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Operating-System Concepts and Implementation | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache English | | | Teilnahmevoraussetzungen none | | |

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr.-Ing. Timo Hönig

Lehrende: Prof. Dr.-Ing. Timo Hönig

Verwendung des Moduls

Lernziele (learning outcomes)

The main learning outcome is a thorough understanding of different operating-system concepts and architectures (e.g., library, monolith, microkernel) and key operating-system components and subsystems. For example, the understanding and practical application of:

- interrupt concepts and implementations
- synchronisation concepts
- pseudo-parallelism in operating systems (e.g., continuations, coroutines, threads, context switches)
- scheduler dimensions of an operating system (e.g., fairness, throughput, preemptive vs. non-preemptive)
- inter-process communication implementations
- memory management and the corresponding memory protection/isolation mechanisms

Orthogonally, important learnings are social skills (e.g., cooperative work in small groups) and the confident use of common software and operating-system development tools (e.g., compilers and debugger).

Inhalt

Building upon the basic knowledge of operating systems and their functionality, this lecture deepens the understanding of fundamental operating-system concepts, architectures, and their implementations. This includes different operating-system architectures for general-purpose operating systems (e.g., Linux, Windows) and special-purpose operating systems (e.g., for embedded or automotive appliances).

Operating-system concepts and components, for example, are:

- memory management
- interrupt subsystem
- input/output subsystem
- processes and threads
- scheduler subsystem
- filesystem implementations
- driver model
- multicore subsystem

Lehrformen

The lecture imparts the theoretical knowledge about operating-system concepts and discusses the respective advantages and disadvantages.

The understanding of those concepts is deepened by implementing operating-system components in small groups as part of the corresponding exercises. This results in a fully functional operating system executable on modern hardware (e.g., x86-64, ARM).

Prüfungsformen

Oral exam (30 minutes)

Voraussetzungen für die Vergabe von Credits

- exam
- successful completion of the exercises

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

Titel des Moduls: Real Time Networks and Systems

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|------------------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Real Time Networks and Systems | | | Kontaktzeit 45 h | Selbststudium 110 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen Keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Steffen Bondorf Lehrende: Prof. Dr.-Ing. Steffen Bondorf | | | | | |
| Verwendung des Moduls Master Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden sollen eine tiefe Kenntnis über Modellierung und Analyse von (harten) Echtzeitsystemen erhalten, insbesondere deterministische Warteschlangentheorie (Networking) sowie Scheduling (Networks, Systems). | | | | | |
| Inhalt Das Scheduling von geteilten Ressourcen (z.B. Forwarding in Netzwerken, CPU-Zeit von lokalen Systemen) spielt eine zentrale Rolle für die Leistungsfähigkeit moderner Systeme. Anforderungen an die Dauer von Datenübertragungen bzw. Berechnungen werden in Echtzeitsystemen in Form von (harten) Deadlines gestellt. Das Einhalten dieser Deadlines muss nachgewiesen bzw. (konstruktiv) garantiert werden. Hierzu stellt die Vorlesung Modellierungs- und Analysemethoden für (verteilte) Echtzeitsysteme vor. | | | | | |
| Lehrformen Die Vorlesung wird als seminaristischer Unterricht abgehalten, die praktischen Übungen am Rechner können weitere Lehrformen wie Gruppen- und Projektarbeit beinhalten. | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an Übungen | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|-----------------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: System Performance Evaluation | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen System Performance Evaluation | | | Kontaktzeit 45 h | Selbststudium 110 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen kein | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Steffen Bondorf Lehrende: Prof. Dr.-Ing. Steffen Bondorf | | | | | |
| Verwendung des Moduls | | | | | |
| Lernziele (learning outcomes) Die Studierenden sollen eine breite Kenntnis über folgende Themenfelder erlangen: <ul style="list-style-type: none"> • Systemmodellierung und -leistungsanalyse • Entwurf von Experimenten • Messungen • Simulation | | | | | |
| Inhalt Die Veranstaltung „System Performance Evaluation“ vermittelt grundlegende Techniken für Modellierung von Computersystemen und der Quantifizierung ihrer Leistung. Im Vordergrund stehen der Entwurf von Experimenten, Simulation und Warteschlangen-theorie. | | | | | |
| Lehrformen Die Vorlesung wird als seminaristischer Unterricht abgehalten, die praktischen Übungen am Rechner können weitere Lehrformen wie Gruppen- und Projektarbeit beinhalten. | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an Übungen | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Titel des Moduls: Geometrische Algorithmen | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Geometrische Algorithmen | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Maike Buchin Lehrende: Prof. Dr. Maike Buchin | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende grundlegende geometrische Algorithmen und Datenstrukturen • können Studierende Algorithmen nach dem Sweep-Paradigma analysieren und entwerfen • können Studierende inkrementelle Algorithmen entwerfen und analysieren, insbesondere randomisiert inkrementelle Algorithmen • können Studierende geometrische Algorithmen nach dem Teile-und-Herrsche Prinzip analysieren und entwerfen • können Studierende für Bereichsanfragen geeignete Datenstrukturen aussuchen | | | | | |
| Inhalt Die Algorithmische Geometrie beschäftigt sich mit dem Entwurf und der Analyse von Algorithmen und Datenstrukturen für geometrische Probleme. Dazu werden zunächst einige grundlegende Probleme betrachtet, wie das Berechnen der konvexen Hülle einer Punktmenge, der Schnittpunkte einer Menge von Strecken oder einer Triangulierung eines einfachen Polygons. Anschließend sehen wir Algorithmen zum Berechnen bekannter geometrische Strukturen, wie das Voronoi-Diagramm, die Delaunay-Triangulierung und Arrangements. Ebenfalls betrachten wir Datenstrukturen für effiziente Anfragen auf geometrischen Daten, wie Rangetrees, kd-Bäume und Quadrees. Dabei kommen vor allem drei Arten von Algorithmen zum Einsatz: inkrementell, teile-und-herrsche, und sweep. Manche von diesen treten als randomisierte Algorithmen auf. | | | | | |
| Lehrformen Vorlesung als kombinierter Folien- und Tafelvortrag und zugehörige Übungen. | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Prüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik 5/97: M.Sc. Informatik | | | | | |

| | | | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Information Theory | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Information Theory | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit M.Sc. Informatik M.Sc. IT-Sicherheit / Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in information theory. After successful completion of this course, you will know the mathematical model of information theory, how to design and analyze algorithms for a variety of information processing tasks, and how to implement them in Python. You will have independently read about a topic in information theory and presented it to your peers. You will be prepared for an advanced course or a research or thesis project in this area. Please see the course homepage for a precise list of learning objectives. | | | | | |

Inhalt

This course will give an introduction to information theory – the mathematical theory of information. Ever since its inception, information theory has had a profound impact on society. It underpins important technological developments, from reliable memories to mobile phone standards, and its versatile mathematical toolbox has found use in computer science, machine learning, physics, electrical engineering, mathematics, and many other disciplines.

Starting from probability theory, we will discuss how to mathematically model information sources and communication channels, how to optimally compress information, and how to design error-correcting codes that allow us to reliably communicate over noisy communication channels. We will also see how techniques used in information theory can be applied more generally to make predictions from noisy data.

Tentative syllabus:

- Welcome, Introduction to Information Theory
- Probability Theory Refresher
- Numerical Random Variables, Convexity and Concavity, Entropy
- Symbol Codes: Lossless Compression, Huffman Algorithm
- Block Codes: Shannon's Source Coding Theorem, its Proof, and Variations
- Stream Codes: Lempel-Ziv Algorithm
- Stream Codes: Arithmetic Coding
- Joint Entropies & Communication over Noisy Channels
- Shannon's Noisy Coding Theorem
- Proof of the Noisy Coding Theorem
- Proof of the Converse, Shannon's Theory vs Practice
- Reed-Solomon Codes
- Message Passing for Decoding and Inference, Outlook
- Student Presentations

Please see the course homepage https://qi.rub.de/it_ss23 for more information.

Lehrformen

Lecture with Exercise

Prüfungsformen

Wird noch nachgeliefert

Voraussetzungen für die Vergabe von Credits

Passed Exam.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

5/97: M.Sc. Informatik

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

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

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Komplexitätstheorie
Complexity Theory

| | | | | | |
|---------------------------------------------------|------------------------|--------------------------|---------------------------------|-----------------------------------------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Unregelmäßig (i.d.R. Sommersemester) | Dauer 1 Semester |
| Lehrveranstaltungen Komplexitätstheorie | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Thomas Zeume
 Lehrende: Prof. Thomas Zeume

Verwendung des Moduls

M.Sc. Informatik
 M.Sc. Angewandte Informatik

Lernziele (learning outcomes)

Die Studierenden lernen, algorithmische Probleme bezüglich ihrer Komplexität einzuordnen und so geeignete algorithmische Techniken zu ihrer Lösung zu identifizieren. Sie können insbesondere algorithmische Methoden für NP-vollständige Probleme anwenden. Sie können mit unterschiedlichen Berechnungsmodellen umgehen und sind in der Lage, einfache Aussagen über sie zu beweisen. Sie lernen im Diskurs eigene und fremde Lösungsansätze zu bewerten.

Inhalt

Die Komplexitätstheorie untersucht und klassifiziert Berechnungsprobleme bezüglich ihrer algorithmischen Schwierigkeit. Ziel ist es, den inhärenten Ressourcenverbrauch bezüglich verschiedener Ressourcen wie Rechenzeit oder Speicherplatz zu bestimmen, und Probleme mit ähnlichem Ressourcenverbrauch in Komplexitätsklassen zusammenzufassen. Die bekanntesten Komplexitätsklassen sind sicherlich P und NP, die die in polynomieller Zeit lösbaren bzw. verifizierbaren Probleme umfassen. Die Frage, ob P und NP verschieden sind, wird als eine der bedeutendsten offenen Fragen der theoretischen Informatik, ja sogar der Mathematik, angesehen. P und NP sind jedoch nur zwei Beispiele von Komplexitätsklassen. Andere Klassen ergeben sich unter anderem bei der Untersuchung der des benötigten Speicherplatzes, der effizienten Parallelisierbarkeit von Problemen, der Lösbarkeit durch zufallsgesteuerte Algorithmen, und der approximativen Lösbarkeit von Problemen. Die Vorlesung hat das Ziel, einen breiten Überblick über die grundlegenden Konzepte und Resultate der Komplexitätstheorie zu geben:

- Klassische Resultate für Platz- und Zeitkomplexitätsklassen: z.B. die Korrespondenz zwischen Spielen und Speicherplatz-Beschränkungen, der Nachweis, dass sich mit mehr Platz oder Zeit auch mehr Probleme lösen lassen, weitere grundlegende Beziehungen zwischen Zeit- und Platzbasierten Klassen, und die Komplexitätswelt zwischen NP und PSPACE
- Grundzüge der Komplexitätstheorie paralleler, zufallsbasierter und approximativer Algorithmen
- Einführung in ausgewählte neuere Themen: Komplexitätstheorie des interaktiven Rechnens, des probabilistischen Beweisens und Fine-grained Complexity.

Lehrformen

Vorlesung mit Übungen

Prüfungsformen

Abschlussprüfung; mündlich, 20-30min

Voraussetzungen für die Vergabe von Credits

Bestandene mündliche Modulabschlussprüfung

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

9/105: M.Sc. Angewandte Informatik

9/97: M.Sc. Informatik

Titel des Moduls: Model Checking

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Model Checking | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Dr. Nils Vortmeier Marko Schmellenkamp | | | | | |
| Verwendung des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Mathematik | | | | | |
| Lernziele (learning outcomes) Die Studierenden lernen wie sich verteilte Systeme durch Transitionssysteme modellieren und Eigenschaften in logischen Spezifikationssprachen wie LTL und CTL spezifizieren lassen. Sie sollen elementare Algorithmen zur Überprüfung von Eigenschaften in Transitionssystemen kennenlernen. Sie sollen ein Verständnis für die Möglichkeiten und Grenzen des Model Checkings entwickeln, und in die Lage versetzt werden, sich eigenständig mit fortgeschrittenen Methoden des Model Checkings auseinanderzusetzen. | | | | | |
| Inhalt Wie kann die Korrektheit von Software und Hardware formal überprüft werden? Im Model Checking werden Software- und Hardware-Module durch Transitionssysteme formalisiert; gewünschte Eigenschaften mit Hilfe logischer Formalismen formal beschrieben; und mit Hilfe von Algorithmen automatisiert überprüft, ob ein Transitionssystem eine formal spezifizierte Eigenschaft besitzt. In dieser Veranstaltung werden die theoretischen Grundlagen des Model Checkings vermittelt, mit einem Fokus auf logik-basierten Spezifikationssprachen. Die Spezifikationssprachen LTL und CTL werden eingeführt, ihre Ausdrucksstärke untersucht, und die wichtigsten algorithmischen Ansätze für das Model Checking vorgestellt. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Abschlussprüfung; mündliche Prüfung (20-30min) oder schriftliche Klausur (120min) in Abhängigkeit der Teilnehmerzahl | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulprüfung. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/170: B.Sc. Informatik [PO 22]

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

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

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

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Quantum Information and Computation | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Quantum Information and Computation | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache German or Englisch (depends on audience) | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in quantum information and computation. After successful completion of this course, you will know the theoretical model of quantum information and computation, how to generalize computer science concepts to the quantum setting, how to design and analyze quantum algorithms and protocols for a variety of computational problems, and how to prove complexity theoretic lower bounds. You will be prepared for an advanced course or a research or thesis project in this area. | | | | | |
| Inhalt This course will give an introduction to quantum information and quantum computation from the perspective of theoretical computer science. Topics to be covered will likely include: <ul style="list-style-type: none"> • Fundamentals of quantum computing: quantum bits, states and operations • The power of quantum entanglement: nonlocal games • Entanglement as a resource: superdense coding and teleportation • Quantum circuit model of computation • Quantum computing with oracles: Deutsch-Jozsa, Bernstein-Vazirani, Simon • Quantum Fourier transform and phase estimation • Shor's factoring algorithm • Grover's search algorithm and beyond: how to solve SAT on a quantum computer? • From no cloning to quantum money: a peek at quantum cryptography <p>The course should be of interest to students of computer science, mathematics, physics, and related disciplines. Students interested in a BSc or MSc project in quantum information, computing, cryptography, etc. are particularly encouraged to participate.</p> | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Written Exam (120 Minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Blockchain Security and Privacy

| | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Blockchain security and privacy | | | Kontaktzeit 4 SWS | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Ghassan Karame Lehrende: | | | | | |
| Verwendung des Moduls Master IT-Sicherheit Master Informatik | | | | | |
| Lernziele (learning outcomes) Upon completion of this course, students are expected to be able to: <ol style="list-style-type: none">1. Reason about the security and privacy definitions of open payment systems.2. Explain the security of PoW blockchains in light of the state of the art reported attacks.3. Reason about possible network security and cryptographic countermeasures to deter attacks on blockchains.4. Explain best security/privacy practices to strengthen the security of existing blockchains, and extract relevant lessons for the design of next-generation blockchain technologies. | | | | | |
| Inhalt The main objective of the course is to provide a comprehensive overview of the security and privacy of blockchain technologies. Course participants will be also introduced to the basic security and privacy provisions of existing popular currencies, and will be exposed to the state-of-the-art attacks and threats reported against existing systems/deployments. The participants will also reason on the effectiveness of combining network-level security primitives, with novel cryptographic primitives to deter attacks on payment systems. | | | | | |
| Lehrformen Übung mit Vorlesung | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulklausur. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

Titel des Moduls: Kryptographie auf hardwarebasierten Plattformen

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Kryptographie auf hardwarebasierten Plattformen | | | Kontaktzeit 4 SWS | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Tim Güneysu Lehrende: | | | | | |
| Verwendung des Moduls Bachelor IT-Sicherheit Master Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden erlernen die Konzepte der problemorientierten Hardwareentwicklung mit abstrakten Hardwarebeschreibungssprachen (VHDL) sowie die Simulation von Hardwareentwicklungen auf rekonfigurierbaren Plattformen. Sie beherrschen (a) Standard- und (b) Optimierungstechniken für kryptographische Systeme auf Hardwareebene und können (c) vollständige Implementierungen von symmetrischen und asymmetrischen Kryptosystemen auf modernen FPGA-Plattformen realisieren. | | | | | |
| Inhalt Kryptographische Systeme stellen aufgrund ihrer Komplexität insbesondere an kleine Prozessoren und eingebettete Systeme hohe Anforderungen. In Kombination mit dem Anspruch von hohem Datendurchsatz bei geringsten Hardwarekosten ergeben sich hier für den Entwickler grundlegende Probleme, die in dieser Vorlesung beleuchtet werden sollen. Die Vorlesung behandelt die interessantesten Aspekte, wie man aktuelle kryptographische Verfahren auf praxisnahen Hardwaresystemen implementiert. Dabei werden Kryptosysteme wie die Blockchiffre AES, die Hashfunktionen SHA-1 sowie asymmetrische Systeme RSA und ECC behandelt. Weiterhin werden auch spezielle Hardwareanforderungen wie beispielsweise der Erzeugung echten Zufalls (TRNG) sowie der Einsatz von Physically Unclonable Functions (PUF) besprochen. Die effiziente Implementierung dieser Kryptosysteme, insbesondere in Bezug auf die Optimierung für Hochgeschwindigkeit, wird auf modernen FPGAs besprochen und in praktischen Übungen mit Hilfe der Hardwarebeschreibungssprache VHDL umgesetzt. Vorlesungsbegleitend wird ein Moodle-Kurs angeboten, der zusätzliche Inhalte sowie die praktischen Übungen bereithält. | | | | | |
| Lehrformen Vorlesungen und Übungen | | | | | |
| Prüfungsformen schriftlich, 120 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulklausur (100 Prozent der Modulabschlussnote). Durch die erfolgreiche Teilnahme am Übungsbetrieb können bis zu 10 Prozent Bonuspunkte erworben werden, die auf das Ergebnis der Modulklausur angerechnet werden können. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| Titel des Moduls: Kryptographie auf hardwarebasierten Plattformen | | | | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Kryptographie auf hardwarebasierten Plattformen (212019) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Tim Güneysu Lehrende: Prof. Dr.-Ing. Tim Güneysu | | | | | |
| Verwendung des Moduls B.Sc. IT-Sicherheit / Informationstechnik B.Sc. Angewandte Informatik M.Sc. IT-Sicherheit / Netze und Systeme M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden kennen die Konzepte der praxisnahen Hardwareentwicklung mit abstrakten Hardwarebeschreibungssprachen (VHDL) und die Simulation von Hardwareschaltungen auf FPGAs. Sie beherrschen Standardtechniken der hardwarenahen Prozessorentwicklung und sind zur Implementierung von symmetrischen und asymmetrischen Kryptosystemen auf modernen FPGA-Systemen in der Lage. | | | | | |
| Inhalt Kryptographische Systeme stellen aufgrund ihrer Komplexität ins- besondere an kleine Prozessoren und eingebettete Systeme hohe Anforderungen. In Kombination mit dem Anspruch von hohem Datendurchsatz bei geringsten Hardwarekosten ergeben sich hier für den Entwickler grundlegende Probleme, die in dieser Vorlesung beleuchtet werden sollen. Die Vorlesung behandelt die interessantesten Aspekte, wie man aktuelle kryptographische Verfahren auf praxisnahen Hardwaresystemen implementiert. Dabei werden Kryptosysteme wie die Blockchiffre AES, die Hashfunktionen SHA-1 sowie asymmetrische Systeme RSA und ECC behandelt. Weiterhin werden auch spezielle Hardwareanforderungen wie beispielsweise der Erzeugung echten Zufalls (TRNG) sowie der Einsatz von Physically Unclo- nable Functions (PUF) besprochen. Die effiziente Implementierung dieser Kryptosysteme, insbesondere in Bezug auf die Optimierung für Hochgeschwindigkeit, wird auf modernen FPGAs besprochen und in praktischen Übungen mit Hilfe der Hardwarebeschreibungssprache VHDL umgesetzt. Vorlesungsbegleitend wird ein Moodle-Kurs angeboten, der zusätzliche Inhalte sowie die praktischen Übungen bereithält. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung; Durch die erfolgreiche Teilnahme am Übungsbetrieb können bis zu 10 Prozent Bonuspunkte erworben werden, die auf das Ergebnis der Modulklausur angerechnet werden können. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22]

5/96: M.Sc. IT-Sicherheit / Netze und Systeme [PO 20]

5/97: M.Sc. Informatik

Titel des Moduls: Model Checking

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Model Checking | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Dr. Nils Vortmeier Marko Schmellenkamp | | | | | |
| Verwendung des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Mathematik | | | | | |
| Lernziele (learning outcomes) Die Studierenden lernen wie sich verteilte Systeme durch Transitionssysteme modellieren und Eigenschaften in logischen Spezifikationssprachen wie LTL und CTL spezifizieren lassen. Sie sollen elementare Algorithmen zur Überprüfung von Eigenschaften in Transitionssystemen kennenlernen. Sie sollen ein Verständnis für die Möglichkeiten und Grenzen des Model Checkings entwickeln, und in die Lage versetzt werden, sich eigenständig mit fortgeschrittenen Methoden des Model Checkings auseinanderzusetzen. | | | | | |
| Inhalt Wie kann die Korrektheit von Software und Hardware formal überprüft werden? Im Model Checking werden Software- und Hardware-Module durch Transitionssysteme formalisiert; gewünschte Eigenschaften mit Hilfe logischer Formalismen formal beschrieben; und mit Hilfe von Algorithmen automatisiert überprüft, ob ein Transitionssystem eine formal spezifizierte Eigenschaft besitzt. In dieser Veranstaltung werden die theoretischen Grundlagen des Model Checkings vermittelt, mit einem Fokus auf logik-basierten Spezifikationssprachen. Die Spezifikationssprachen LTL und CTL werden eingeführt, ihre Ausdrucksstärke untersucht, und die wichtigsten algorithmischen Ansätze für das Model Checking vorgestellt. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Abschlussprüfung; mündliche Prüfung (20-30min) oder schriftliche Klausur (120min) in Abhängigkeit der Teilnehmerzahl | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulprüfung. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/170: B.Sc. Informatik [PO 22]

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

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

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

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

Titel des Moduls: Physical Attacks and Countermeasures

| | | | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 Stunden | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Physical Attacks and Countermeasures | | | Kontaktzeit 4 SWS | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Amir Moradi Lehrende: | | | | | |
| Verwendung des Moduls Master IT-Sicherheit Master Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden <ul style="list-style-type: none">• verstehen wie und warum physikalische Angriffe funktionieren.• sind in der Lage Messdaten anhand der erlernten Methoden auszuwerten und die Sicherheit einer Implementierung zu bewerten.• erkennen die Gefahr von physikalischen Angriffen für Implementierungen von kryptographischen Algorithmen.• kennen mögliche Gegenmaßnahmen und wissen wie diese anzuwenden sind, um ein System gegen physikalische Angriffe zu schützen. | | | | | |
| Inhalt Moderne kryptographische Algorithmen bieten ausreichend Schutz gegen die bekannten mathematischen und kryptanalytischen Angriffe. In der Praxis werden diese Algorithmen für sicherheitskritische Anwendungen auf verschiedenen Plattformen implementiert. Dies geschieht sowohl als Programmcode (Software) als auch mit logischen Elementen/Schaltungen (Hardware). Der physikalische Zugang zu kryptographischen Implementierungen (z.B., eine Smartcard oder ein Smartphone, welche zum Bezahlen benutzt werden), in welchen der geheime Schlüssel eingebettet ist, hat zur Entstehung einer neuen Klasse von Angriffen, genannt physikalische Angriffe, geführt. Diese Angriffe zielen darauf ab den geheimen Schlüssel, welcher vom kryptographischen Algorithmus benutzt wird, zu extrahieren. Ein erfolgreicher physikalischer Angriff deutet nicht auf Schwächen im Algorithmus sondern auf Schwachstellen in der Implementierung hin. Daher müssen bereits in der Entwicklungsphase von kryptographischen Implementierungen physikalische Angriffe als potentielles Risiko berücksichtigt und bestmöglich verhindert werden. Das Ziel dieser Lehrveranstaltung ist es einen Überblick über bekannte physikalische Angriffe und deren Gegenmaßnahmen zu geben. Im ersten Teil der Vorlesung werden die verschiedenen Angriffstypen eingeführt, während im zweiten Teil der Fokus auf Gegenmaßnahmen liegt. | | | | | |
| Lehrformen | | | | | |
| Prüfungsformen Schriftliche Prüfung (120 Minuten) | | | | | |

Voraussetzungen für die Vergabe von Credits

Projektbasiertes Arbeiten ist ein großer Teil der Lehrveranstaltung. Zusätzlich zu einer schriftlichen Prüfung gibt es wöchentliche Projektarbeiten (Hausaufgaben) und ein abschließendes Seminar. Alle Teile müssen individuell bearbeitet werden, sind bewertet und gehen in die Endnote ein. Für das erfolgreiche Bestehen des Kurses muss die Klausur mit mindestens 50

Wöchentliche Projektarbeiten (Hausaufgaben): 30

Klausur: 60

Abschließendes Seminar: 20

Dies ergibt eine Summe von 110

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Quantum Information and Computation | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Quantum Information and Computation | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache German or Englisch (depends on audience) | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in quantum information and computation. After successful completion of this course, you will know the theoretical model of quantum information and computation, how to generalize computer science concepts to the quantum setting, how to design and analyze quantum algorithms and protocols for a variety of computational problems, and how to prove complexity theoretic lower bounds. You will be prepared for an advanced course or a research or thesis project in this area. | | | | | |
| Inhalt This course will give an introduction to quantum information and quantum computation from the perspective of theoretical computer science. Topics to be covered will likely include: <ul style="list-style-type: none"> • Fundamentals of quantum computing: quantum bits, states and operations • The power of quantum entanglement: nonlocal games • Entanglement as a resource: superdense coding and teleportation • Quantum circuit model of computation • Quantum computing with oracles: Deutsch-Jozsa, Bernstein-Vazirani, Simon • Quantum Fourier transform and phase estimation • Shor's factoring algorithm • Grover's search algorithm and beyond: how to solve SAT on a quantum computer? • From no cloning to quantum money: a peek at quantum cryptography <p>The course should be of interest to students of computer science, mathematics, physics, and related disciplines. Students interested in a BSc or MSc project in quantum information, computing, cryptography, etc. are particularly encouraged to participate.</p> | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Written Exam (120 Minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Autonomous Vehicles and Artificial Intelligence

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|------------------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester 5 | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Vehicles and Artificial Intelligence | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 25 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Thorsten Berger Lehrende: Prof. Dr. Thorsten Berger, Dr. Sven Peldszus | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit B.Sc. Angewandte Informatik M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit/Informationstechnik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none">• Understanding requirements on autonomous vehicles• Understanding the architecture of autonomous vehicles• Ability to build a self-driving car with ROS2• Understanding and applying quality assurance for autonomous vehicles | | | | | |
| Inhalt <p>Autonomous driving is the future of individual mobility and all major manufacturers are working on fully autonomous vehicles. While there are robust and good solutions for the individual problems in autonomous driving, the main challenge lies in their integration. Altogether, an autonomous vehicle's software is the biggest problem. Therefore, the key in self-driving vehicles is about getting the software right. In this course, we will investigate the different aspects of self-driving vehicles as well as the importance and application of artificial intelligence in this domain. The course will primarily focus on the following topics:</p> <ul style="list-style-type: none">• Requirements on autonomous vehicles• Architecture of autonomous vehicles• Operation systems and frameworks for robotic systems• Specification and Implementation of autonomous vehicles based on ROS2• Artificial intelligence for autonomous vehicles• Simulation of autonomous vehicles &#8729; Localization and perception• Mission planning• Quality assurance for autonomous vehicles <p>In the course's lecture, we provide the required theoretical background and practically apply the course's content in exercises by building a self-driving robot.</p> | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Mündliche Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an den Übungen | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

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

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

5/ 97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

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

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

Titel des Moduls: High-Performance Computing on Clusters

| | | | | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Clusters | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program and employ parallel computing clusters. Theoretical properties of distributed-memory systems and programming patterns are conveyed as well as the practical implementation. | | | | | |
| Inhalt The lecture deals with the parallelization on cluster computers. Distributed-memory programming concepts (MPI) are introduced and best-practice implementation is presented based on applications from scientific computing including the finite element method and machine learning. Special attention is paid to scalable solvers for systems of equations on distributed memory systems, focusing on iterative schemes such as simple splitting methods (Richardson, Jacobi, Gauß-Seidel, SOR), Krylov-methods (Gradient descent, CG, BiCGStab) and, in particular, the multigrid method. The mathematical foundations for iterative solvers are reviewed, suitable object-oriented interface structures are developed and an implementation of these solvers for modern parallel computer architectures is developed. Numerical experiments and self-developed software implementations are used to discuss and illustrate the theoretical results. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Written Exam (120 minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: High-Performance Computing on Multi- and Manycore Processors

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Multi- and Manycore Processors | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program multi- and manycore processors employing multi-threaded execution and handling shared-memory access patterns. Theoretical properties are conveyed as well as practical implementation. Via presentations of selected topics, students attain the ability to survey and acquire knowledge on advanced scientific topics independently and are qualified to illustrate such topics in the form of a presentation and numerical examples. | | | | | |
| Inhalt The lecture addresses parallelization for multi- and manycore processors. Thread-based programming concepts (pthreads, C++11 threads, OpenMP, OpenCL) are introduced and bestpractice implementation aspects are highlighted based on applications from scientific computing. In the first part, the lecture provides an overview on relevant data structures, solver techniques and programming patterns from scientific computing. An introduction to multithreading programming on multicore systems is then provided with special attention to shared-memory aspects. Parallelization patterns are discussed and highlighted. Numerical experiments and self-developed software implementations are used to discuss and illustrate the presented content. In the second part, students are assigned advanced topics for sharedmemory computation from the engineering science including finite element methods and artificial intelligence. Based on a scientific paper, students present their topic to the lecture audience in form of a beamer presentation and numerical illustrations. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Semesterbegleitend; Hausarbeit | | | | | |
| Voraussetzungen für die Vergabe von Credits mit mindestens ausreichend bewertete Hausarbeit | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Autonomous Robotics: Action, Perception, Cognition

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Robotics: Action, Perception, Cognition (211048) | | | Kontaktzeit 45 h | Selbststudium 135 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Gregor Schöner Lehrende: Prof. Dr. Gregor Schöner | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none">• are familiar with the concepts in dynamical systems theory and can use them practically,• know the mathematical models of movement generation• have practice in reading and writing academic research papers. | | | | | |
| Inhalt Autonomous robotics is an interdisciplinary research field in which embodied systems equipped with their own sensors and with actuators generate behavior that is not completely preprogrammed. Autonomous robotics thus entails perception, movement generation, as well as core elements of cognition such as making decisions, planning, and integrating multiple constraints. This course touches on various approaches to this interdisciplinary problem. In the first half of the course, the main emphasis will be on dynamical systems methods for generating movement in vehicles. The main focus of the course is, however, on solutions to autonomous movement generation that are inspired by analogies with how nervous systems generate movement. In fact, the second half of the course will review core problems in human movement science, including the degree of freedom problem, coordination, motor control, and the reflex control of muscles. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Oral Examination; Exercises for bonus-points | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed oral examination | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Autonomous Vehicles and Artificial Intelligence

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|------------------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester 5 | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Vehicles and Artificial Intelligence | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 25 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Thorsten Berger Lehrende: Prof. Dr. Thorsten Berger, Dr. Sven Peldszus | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit B.Sc. Angewandte Informatik M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit/Informationstechnik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none">• Understanding requirements on autonomous vehicles• Understanding the architecture of autonomous vehicles• Ability to build a self-driving car with ROS2• Understanding and applying quality assurance for autonomous vehicles | | | | | |
| Inhalt <p>Autonomous driving is the future of individual mobility and all major manufacturers are working on fully autonomous vehicles. While there are robust and good solutions for the individual problems in autonomous driving, the main challenge lies in their integration. Altogether, an autonomous vehicle's software is the biggest problem. Therefore, the key in self-driving vehicles is about getting the software right. In this course, we will investigate the different aspects of self-driving vehicles as well as the importance and application of artificial intelligence in this domain. The course will primarily focus on the following topics:</p> <ul style="list-style-type: none">• Requirements on autonomous vehicles• Architecture of autonomous vehicles• Operation systems and frameworks for robotic systems• Specification and Implementation of autonomous vehicles based on ROS2• Artificial intelligence for autonomous vehicles• Simulation of autonomous vehicles &#8729; Localization and perception• Mission planning• Quality assurance for autonomous vehicles <p>In the course's lecture, we provide the required theoretical background and practically apply the course's content in exercises by building a self-driving robot.</p> | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Mündliche Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an den Übungen | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

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

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

5/ 97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

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

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

| | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Computational Neuroscience: Neural Dynamics | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Neural Dynamics (212005) | | | Kontaktzeit 45 h | Selbststudium 135 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Gregor Schöner Lehrende: Prof. Dr. Gregor Schöner | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Gain experience in interdisciplinarity bridging computer science and cognitive science. • Learn the concepts and methods of nonlinear dynamical systems in a concrete applied context. • Improve familiarity with methods of quantitative natural science, including measurement, graphing observables as a function of experimental control parameters and using models to interpret data. • Read scientific literature. | | | | | |
| Inhalt This course provides an introduction into the theoretical cognitive and functional neurosciences from a particular theoretical vantage point, the dynamical systems approach. This approach emphasizes the evolution in time of behavioral and neural patterns as the basis of their analysis and synthesis. Dynamic stability, a concept shared with the classical biological cybernetics framework, is one cornerstone of the approach. Instabilities (or bifurcations) extend this framework and provide a basis for understanding flexibility, task specific adjustment, adaptation, and learning. The course includes tutorial modules that provide mathematical foundations. Theoretical concepts are expounded in reference to a number of experimental model systems which include the coordination of movement, postural stability, the perception of motion, and elementary forms of embodied cognition. In the spirit of Braitenberg's "synthetic psychology", autonomous robots are used to illustrate some of the ideas. Exercises are integrated into the lectures. They consist of elementary mathematical exercises, the design of (thought) experiments and their analysis, and the design of simple artificial systems, all on the basis of the theoretical framework exposed in the main lectures. One exercise takes the form of an essay for which participants read a scientific paper and answer questions in a longer illustrated text. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Oral or written exam, Bonus points for the final examination can be achieved by submitting homework and an essay (10 pages) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed oral or written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Computational Neuroscience: Single-Neuron Models

| | | | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Single-Neuron Models | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Robert Schmidt Lehrende: Prof. Dr. Robert Schmidt | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach Abschluss des Moduls können Studierende <ul style="list-style-type: none">• können Studierende Techniken der Computational Neuroscience zur Simulation neuronaler Aktivität anwenden• haben sich Studierende mit verschiedenen Arten von Modellen einzelner Neuronen, ihrer mathematischen Beschreibung und ihren verschiedenen biologischen Abstraktionsebenen vertraut gemacht• haben Studierende die Fähigkeiten zur Modellierung von Neuronen, Synapsen und Schaltkreisen erworben und und können diese Modelle mit Biologie und Berechnung verknüpfen• besitzen Studierende ein Verständnis der biologischen Grundlage für Berechnungen in Neuronen | | | | | |
| Inhalt Dieses Modul beginnt mit einer Einführung in die Neurowissenschaften und die Rolle der Computational Neuroscience. Der nächste Teil des Moduls befasst sich mit biologisch fundierten Modellen einzelner Neuronen, einschließlich Leaky-Integrate-and-Fire- und Leitwert-basierter Neuronen, aber auch mit abstrakteren Modellen neuronaler Aktivität und Spike Trains. Sie werden lernen, wie diese verschiedenen Berechnungsmodelle die zugrunde liegenden biologischen Prozesse in unterschiedlichem Maße beschreiben und vereinfachen. Wir werden im Detail untersuchen, wie diese verschiedenen Neuronenmodelle in numerischen Simulationen verwendet werden können, um Forschungsfragen zur Berechnung in einzelnen Neuronen und Schaltkreisen zu beantworten. In den vorlesungsbegleitenden Übungen werden Sie praktische Erfahrungen mit der Implementierung der verschiedenen Neuronenmodelle in Python, der Durchführung numerischer Simulationen und der Durchführung von Berechnungen im Zusammenhang mit analytischen Lösungen der Modellgleichungen und der Biophysik sammeln. Der Schwerpunkt liegt auf Einzelneuronenmodellen, aber wir werden auch verfügbare Software (z.B. NEST Desktop) nutzen, um zu untersuchen, wie Einzelneuronenmodelle in Simulationen von neuronalen Netzwerken integriert werden können. Während der Schwerpunkt des Moduls auf methodischen Fragen liegt und darauf, wie Modelle auf jeder Ebene aufgebaut, getestet und validiert werden können, werden wir auch Verbindungen zu bestimmten Gehirnregionen herstellen, um die Modelle zu motivieren und zu veranschaulichen. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Schriftliche Prüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulklausur. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/97: M.Sc. Informatik

6/105: M.Sc. Angewandte Informatik

Titel des Moduls: Computational Neuroscience: Vision and Memory

| | | | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Vision and Memory (211049) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Cognitive Science | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none">• know basic neurobiological facts about the visual system and the hippocampus,• know a number of related models and methods in computational neuroscience,• understand the mathematics of these methods,• can communicate about all this in English. | | | | | |
| Inhalt This lecture covers basic neurobiology and models of selforganization in neural systems, in particular addressing Learning and self-organization <ul style="list-style-type: none">• Hebbian Learning• Neural learning dynamics and constrained optimization• Dynamic field theory Vision <ul style="list-style-type: none">• Receptive fields• Neural maps• Hippocampus• Navigation• Episodic memory• Hopfield Network | | | | | |
| Lehrformen This course is given with the flipped/inverted classroom concept. First, the students work through online material by themselves. In the lecture time slot we then discuss the material, find connections to other topics, ask questions and try to answer them. In the tutorial time slot the newly acquired knowledge is applied to analytical exercises and thereby deepened. I encourage all students to work in teams during self-study time as well as in the tutorial. | | | | | |
| Prüfungsformen The course is concluded with a digital written exam. | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Deep Learning

| | | | | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Deep Learning | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. IT-Sicherheit / Informationstechnik M.Sc. IT-Sicherheit / Netze und Systeme M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Vorlesung hat das Ziel, einen Einblick in dieses Gebiet zu vermitteln. Zu Beginn werden die grundlegenden Begriffe und Konzepte des maschinellen Lernens eingeführt. Im weiteren Verlauf wird auf verschiedene neuronale Netze, Gradienten-basierte Optimierungsverfahren und generative Modelle eingegangen. | | | | | |
| Inhalt Deep Learning ist ein Untergebiet des maschinellen Lernens, welches in den letzten Jahren zu Durchbrüchen in zahlreichen Anwendungsgebieten (wie z.B. in der Objekt- und Spracherkennung und der maschinellen Übersetzung) geführt hat. Deep Learning Methoden finden unter anderem Anwendung im Bereich IT Security | | | | | |
| Lehrformen Vorlesung und Übung | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/91: M.Sc. IT-Sicherheit / Informationstechnik [PO 22] 5/84: M.Sc. IT-Sicherheit / Informationstechnik [PO 20] 5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22] 5/96: M.Sc. IT-Sicherheit / Netze und Systeme [PO 20] 5/105: M.Sc. Angewandte Informatik 5/ 97: M.Sc. Informatik | | | | | |

| | | | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Information Theory | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Information Theory | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit M.Sc. Informatik M.Sc. IT-Sicherheit / Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in information theory. After successful completion of this course, you will know the mathematical model of information theory, how to design and analyze algorithms for a variety of information processing tasks, and how to implement them in Python. You will have independently read about a topic in information theory and presented it to your peers. You will be prepared for an advanced course or a research or thesis project in this area. Please see the course homepage for a precise list of learning objectives. | | | | | |

Inhalt

This course will give an introduction to information theory – the mathematical theory of information. Ever since its inception, information theory has had a profound impact on society. It underpins important technological developments, from reliable memories to mobile phone standards, and its versatile mathematical toolbox has found use in computer science, machine learning, physics, electrical engineering, mathematics, and many other disciplines.

Starting from probability theory, we will discuss how to mathematically model information sources and communication channels, how to optimally compress information, and how to design error-correcting codes that allow us to reliably communicate over noisy communication channels. We will also see how techniques used in information theory can be applied more generally to make predictions from noisy data.

Tentative syllabus:

- Welcome, Introduction to Information Theory
- Probability Theory Refresher
- Numerical Random Variables, Convexity and Concavity, Entropy
- Symbol Codes: Lossless Compression, Huffman Algorithm
- Block Codes: Shannon's Source Coding Theorem, its Proof, and Variations
- Stream Codes: Lempel-Ziv Algorithm
- Stream Codes: Arithmetic Coding
- Joint Entropies & Communication over Noisy Channels
- Shannon's Noisy Coding Theorem
- Proof of the Noisy Coding Theorem
- Proof of the Converse, Shannon's Theory vs Practice
- Reed-Solomon Codes
- Message Passing for Decoding and Inference, Outlook
- Student Presentations

Please see the course homepage https://qi.rub.de/it_ss23 for more information.

Lehrformen

Lecture with Exercise

Prüfungsformen

Wird noch nachgeliefert

Voraussetzungen für die Vergabe von Credits

Passed Exam.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

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

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

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

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

5/97: M.Sc. Informatik

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

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

5/105: M.Sc. Angewandte Informatik

| | | | | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Knowledge Graphs | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Knowledge Graphs | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Jun.-Prof. Maribel Acosta Deibe Lehrende: Jun.-Prof. Maribel Acosta Deibe | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) In this lecture, students will learn about the foundations of modelling, querying, publishing, and reasoning over KGs. The topics will be complemented with exercises and Jupyter Notebooks (https://jupyter.org/) to show how KG technologies work in practice. | | | | | |
| Inhalt Knowledge Graphs (KG) allow for representing inter-connected facts or statements annotated with semantics. In KGs, concepts and entities are typically modeled as nodes while their connections are modeled as directed and labeled edges, creating a graph. In recent years, KGs have become core components of modern data ecosystems. KGs, as building blocks of many Artificial Intelligence approaches, allow for harnessing and uncovering patterns from the data. Currently, KGs are used in the data-driven business processes of multinational companies like Google, Microsoft, IBM, eBay, and Facebook. Furthermore, thousands of KGs are openly available on the web following the Linked Data (https://lod-cloud.net/) principles. The specific topics covered in the lecture are as follows: <ol style="list-style-type: none"> 1. Introduction to Knowledge Graphs 2. The Resource Description Framework (RDF) 3. RDF Schema (RDFS) 4. The SPARQL Query Language 5. Semantics of SPARQL 6. Linked Data: Knowledge Graphs and Ontologies on the Web 7. The Web Ontology Language (OWL) 8. Entailment Regimes 9. Reasoning over Knowledge Graphs 10. Property Graphs 11. Knowledge Graph Applications | | | | | |
| Lehrformen Lecture with Exercise | | | | | |

Prüfungsformen

Written Exam (120 minutes)

Voraussetzungen für die Vergabe von Credits

Passed written Exam.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Machine Learning: Evolutionary Algorithms

| | | | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Evolutionary Algorithms | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tobias Glasmachers Lehrende: | | | | | |
| Verwendung des Moduls Master Angewandte Informatik Master Informatik | | | | | |
| Lernziele (learning outcomes) Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none">• kennen die Teilnehmer die wichtigsten Klassen direkter Optimierungsverfahren und ihre algorithmischen Komponenten,• haben die Teilnehmer ein tiefes Verständnis evolutionärer Algorithmen, insbesondere für kontinuierliche Probleme,• kennen die Teilnehmer eine Reihe spezifischer Problemschwierigkeiten und die zugehörigen algorithmischen Komponenten, welche diese Adressieren,• können die Teilnehmer elementare Laufzeitanalysen durchführen und verstehen die wichtigsten Konvergenzklassen &#8729; können die Teilnehmer Optimierungsverfahren selbst implementieren und zur Lösung neuer Probleme anwenden. | | | | | |
| Inhalt Breiter Überblick über Optimierungsverfahren. Evolutionäre Optimierungsverfahren für black-box Optimierungsverfahren Algorithmische Komponenten für schlechte Konditionierung, Multimodalität, Rauschen, Nebenbedingungen und Mehrzieloptimierung. Konvergenz und Laufzeitanalyse. | | | | | |
| Lehrformen flipped classroom | | | | | |
| Prüfungsformen Abschlussprüfung: Klausur 90 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik 6/97: M.Sc. Informatik | | | | | |

Titel des Moduls: Machine Learning: Supervised Methods

| | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 Stunden | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Supervised Methods | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße 80 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tobias Glasmachers Lehrende: Prof. Dr. Tobias Glasmachers | | | | | |
| Verwendung des Moduls Master Informatik Master Angewandte Informatik Master IT-Sicherheit/Netze und Systeme | | | | | |
| Lernziele (learning outcomes) Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none">• verstehen die Teilnehmer die Grundlagen der statistischen Lerntheorie• kennen die Teilnehmer die wichtigsten Algorithmen des überwachten statistischen Lernens und können diese auf Lernprobleme anwenden,• kennen die Teilnehmer Stärken und Beschränkungen verschiedenen Lernverfahren,• können die Teilnehmer Standardsoftware zum maschinellen Lernen zur Lösung neuer Probleme einsetzen. | | | | | |
| Inhalt Grundlagen der statistischen Lerntheorie, Querschnitt der wichtigsten Algorithmen des maschinellen Lernens, konkrete Problemlösung mit Standardsoftware | | | | | |
| Lehrformen flipped classroom | | | | | |
| Prüfungsformen Abschlussprüfung; Klausur 90 Minuten, | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik 6/97: M.Sc. Informatik | | | | | |

Titel des Moduls: Machine Learning: Unsupervised Methods

| | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 9 CP | Workload 270 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Unsupervised Methods | | | Kontaktzeit 60 h | Selbststudium 210 h | Gruppengröße 40 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott | | | | | |
| Verwendung des Moduls Master Informatik Master Angewandte Informatik Master Elektro-und Informationstechnik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none">• know a number of important unsupervised learning methods,• can discuss and decide which of the methods are appropriate for a given data set,• understand the mathematics of these methods,• know how to implement and apply these methods in python,• have gained experience in organizing and working in a team,• know problem solving strategies like brain storming,• can communicate about all this in English. | | | | | |
| Inhalt This course covers a variety of shallow unsupervised methods from machine learning such as principal component analysis, independent component analysis, vector quantization, clustering, Bayesian theory and graphical models. | | | | | |
| Lehrformen This course is given in a hybrid of inverted classroom and problem based learning. The course starts with a two-week introduction into unsupervised methods of machine learning, providing an overview. The students then work in groups of about 4 on realistic problems that can be solved with these methods. In the first week of a problem, they develop hypotheses and strategies for a solution and identify which methods they want to learn. Then the course agrees on a method to focus on theoretically, which will then be done in an inverted classroom format. The students then try to solve the problem and present their results in a short talk with slides. Thus the students will not only learn about machine learning but also soft skills. | | | | | |
| Prüfungsformen Semesterbegleitend; The exam is a combination of graded presentations for the problems and graded quizzes for the theory. 40% of the grade come from the average group performance on solving the problems. 10% come from the presentations, taking into account slides and presentation style, this is an individual grade of the presenter. 50% come from a digital quiz about the theory of the methods covered. Thus 60% of the grade are individual, 40% come from the group. In addition you can gain up to 8 bonus points for (i) being voted for as a 'most valuable player (MVP)' on a project and (ii) creating useful quiz questions. | | | | | |
| Voraussetzungen für die Vergabe von Credits Continuous participation and passed exam. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

9/105: M.Sc. Angewandte Informatik

9/97: M.Sc. Informatik

| Titel des Moduls: Statistisches Lernen und Data Mining | | | | | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|------------------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Statistisches Lernen und Data Mining | | | Kontaktzeit 60 h | Selbststudium 90 | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Johannes Lederer Lehrende: Prof. Dr. Johannes Lederer | | | | | |
| Verwendung des Moduls B.Sc. Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls - kennen die Studierenden Standardmethoden der Datenanalyse - verstehen die Studierenden, wann welche Methoden passend sind - sind die Studierenden in der Lage, die Methoden anzuwenden - können die Studierenden die Ergebnisse interpretieren) | | | | | |
| Inhalt In dieser Lehrveranstaltung werden die grundlegenden Methoden der Datenanalyse eingeführt. Dabei werden verschiedene Datentypen berücksichtigt, insbesondere Regressionsdaten und Klassifikationsdaten. Immer werden auch die zu Grunde liegenden statistischen Modelle besprochen. Ebenfalls werden mögliche Anwendungen sowohl im Unterricht als auch in Computer-Übungen vorgestellt. 105 Ziel ist es, den gesamten Verlauf einfacher Datenanalysen zu vermitteln: Datenaufbereitung, statistische Modellbildung, Auswahl einer Methode, Implementierung der Methode, Visualisierung der Resultate und Interpretation. | | | | | |
| Lehrformen Hörsaalvorlesung mit Medienunterstützung, insbesondere Datenanalysen mit dem Computer, Tutorien als seminaristischer Unterricht, zusätzlich Selbststudium mit ergänzend bereitgestellten Materialien und Aufgaben | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung über 90 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/158: B.Sc. Informatik [PO 22] 5/165: B.Sc. Informatik [PO 20] | | | | | |

Titel des Moduls: Theorie des maschinellen Lernens

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Theorie des maschinellen Lernens (211052) | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden werden mit mathematischen Modellen für das maschinelle Lernen vertraut gemacht. Sie erwerben die Fähigkeit, Lernalgorithmen zu beurteilen und zu vergleichen anhand des Grades, in welchem diese (exakt beschriebene) Erfolgskriterien erreichen. Sie erwerben Techniken sowohl zum Design effizienter Lernalgorithmen als zum Nachweis der inhärenten Härte eines Problems. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none">• kennen Studierende die wichtigsten Lernmaschinen (wie zum Beispiel Support Vector Machines und verwandte Modelle),• verstehen Studierende den Unterschied zwischen empirischer und realer Fehlerrate und kennen Techniken zum Umgang mit dem Problem des overfitting der Daten (mit einem zu komplexen Modell),• können Studierende zwischen uniformer und nicht uniformer Lernbarkeit einer Hypothesenklasse unterscheiden und kennen die dazu passenden Theorien und Lernregeln. | | | | | |
| Inhalt Gegenstand der Vorlesung ist die statistik-basierte Theorie des maschinellen Lernens. Insbesondere wird die Methode der strukturierten Risikominimierung vermittelt sowie die ihr zugrunde liegenden statistischen Lehrsätze. Es werden sowohl Techniken zum Entwurf effizienter Lernalgorithmen besprochen als auch informations- oder berechnungstheoretische Barrieren, die bestimmte Lernprobleme als nicht effizient lösbar erscheinen lassen. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen mündliche Modulabschlussprüfung (30 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik 9/97: M.Sc. Informatik | | | | | |

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|-----------------|---------------------------------|--------------------------------------------------|------------------------------------|
| Titel des Moduls: Practical Labs | | | | | |
| Modul-Nr./Code | Credits 3 CP | Workload | Semester | Turnus Jedes Winter und Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Lehrende: | | | | | |
| Verwendung des Moduls | | | | | |
| Lernziele (learning outcomes) | | | | | |
| Inhalt Im Rahmen dieses Bereiches ist ein Laborpraktikum von 3 LP zu absolvieren. Es handelt sich nicht um ein reines Programmierpraktikum, sondern um eine themengebundene Veranstaltung mit wissenschaftlichem Anspruch. | | | | | |
| Lehrformen | | | | | |
| Prüfungsformen | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|-----------------|-----------------------------------------|----------------------|------------------------------------|
| Titel des Moduls: Project | | | | | |
| Modul-Nr./Code | Credits 10 CP | Workload | Semester | Turnus | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache | | | Teilnahmevoraussetzungen none | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Glasmachers Lehrende: | | | | | |
| Verwendung des Moduls Master Informatik | | | | | |
| Lernziele (learning outcomes) | | | | | |
| Inhalt Ein Projekt kann als individuelle Leistung oder als Gruppenarbeit durchgeführt werden. Das forschungsorientierte Projekt im Umfang von 10 LP wird innerhalb eines Semesters absolviert. Es kann direkt auf die Masterarbeit vorbereiten. | | | | | |
| Lehrformen Projektarbeit | | | | | |
| Prüfungsformen Semesterbegleitend | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|-----------------|---------------------------------|--------------------------------------------------|------------------------------------|
| Titel des Moduls: Seminars | | | | | |
| Modul-Nr./Code | Credits 3 CP | Workload | Semester | Turnus Jedes Winter und Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Lehrende: | | | | | |
| Verwendung des Moduls | | | | | |
| Lernziele (learning outcomes) | | | | | |
| Inhalt Im Seminar werden wichtige Kompetenzen wie Literaturarbeit und Präsentationstechniken geübt. Die Studierenden können aus dem breiten Angebot an Seminaren in unterschiedlichen thematischen Bereichen wählen. | | | | | |
| Lehrformen | | | | | |
| Prüfungsformen | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|-----------------|---------------------------------|--------------------------------------------------|------------------------------------|
| Titel des Moduls: Free Elective Moduls | | | | | |
| Modul-Nr./Code | Credits 20 CP | Workload | Semester | Turnus Jedes Winter und Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Lehrende: | | | | | |
| Verwendung des Moduls | | | | | |
| Lernziele (learning outcomes) | | | | | |
| Inhalt Im Rahmen des freien Wahlbereiches sind Lehrveranstaltungen in einem Umfang von mindestens 20 LP zu absolvieren, die frei aus dem Angebot der Universität und der UA-Ruhr zu wählen sind. Dabei können sowohl nichttechnische Lehrveranstaltungen (z.B. Fremdsprachen, Rechts-, Wirtschaft- oder Sozialwissenschaften) als auch technische Lehrveranstaltungen (z.B. Ingenieurwissenschaften, Informatik) gewählt werden. | | | | | |
| Lehrformen | | | | | |
| Prüfungsformen | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

| | | | | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Masterarbeit und Kolloquium (INF) | | | | | |
| Modul-Nr./Code | Credits 30 CP | Workload 900 h | Semester 4 | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 15 h | Selbststudium 885 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen M.Sc. Informatik PO 23: Erfolgreich abgeschlossene Module im Umfang von 70 CP. | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Informatik Lehrende: Lehrende im Studiengang Informatik | | | | | |
| Verwendung des Moduls M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls: <ul style="list-style-type: none"> • können Studierende selbstständig und fristgerecht ein wissenschaftliches Thema bearbeiten von der Recherche bis zur Dokumentation der Resultate • können Studierende geeignete wissenschaftliche Verfahren und Methoden, die sie im Studium kennengelernt haben, auswählen, anwenden und weiterentwickeln, um ein konkretes Problem zu lösen • können Studierende ihre Ergebnisse kritisch mit dem Stand der Forschung vergleichen und evaluieren • können Studierende ihre eigenen Ergebnisse angemessen in Wort und Schrift darstellen. | | | | | |
| Inhalt Die Masterarbeit stellt eine forschungsorientierte, sechsmonatige Arbeit zu einem bestimmten Thema dar und wird im letzten Semester des Studiums geschrieben. Diese hat ein Umfang von 30 Leistungspunkten. Die Masterarbeit wird auf Englisch verfasst. | | | | | |
| Lehrformen Abschlussarbeit | | | | | |
| Prüfungsformen Masterarbeit und Kolloquiumsvortrag | | | | | |
| Voraussetzungen für die Vergabe von Credits Sowohl die Masterarbeit als auch der Kolloquiumsvortrag müssen bestanden sein. Der Anteil der Kolloquiumsnote an der Gesamtnote beträgt 10% | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 30/97 | | | | | |