TUHH
Hamburg
University of
Technology

Module Manual

Master of Science (M.Sc.)

Computer Science

Cohort: Winter Term 2020

Updated: 24th May 2022

Table of Contents

Program description

Content

Core Qualification

Module M0523: Business & Management

Module Responsible	Prof. Matthias Meyer
Admission Requirements	None
Recommended Previous Knowledge	None
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students are able to find their way around selected special areas of management within the scope of business management. • Students are able to explain basic theories, categories, and models in selected special areas of business management. • Students are able to interrelate technical and management knowledge.
Skills	• Students are able to apply basic methods in selected areas of business management. • Students are able to explain and give reasons for decision proposals on practical issues in areas of business management.
Personal Competence	
Social Competence	• Students are able to communicate in small interdisciplinary groups and to jointly develop solutions for complex problems
Autonomy	• Students are capable of acquiring necessary knowledge independently by means of research and preparation of material.
Workload in Hours	Depends on choice of courses
Credit points	6

Courses
Information regarding lectures and courses can be found in the corresponding module handbook published separately.

Module M0524: Non-technical Courses for Master

Module Responsible	Dagmar Richter
Admission Requirements	None
Recommended Previous Knowledge	None
Educational Objectives	After taking part successfully, students have reached the following learning results

Professional Competence

Knowledge

The Nontechnical Academic Programms (NTA)

imparts skills that, in view of the TUHH’s training profile, professional engineering studies require but are not able to cover fully. Self-reliance, self-management, collaboration and professional and personnel management competences. The department implements these training objectives in its **teaching architecture**, in its **teaching and learning arrangements**, in **teaching areas** and by means of teaching offerings in which students can qualify by opting for **specific competences** and a **competence level** at the Bachelor’s or Master’s level. The teaching offerings are pooled in two different catalogues for nontechnical complementary courses.

The Learning Architecture

consists of a cross-disciplinarily study offering. The centrally designed teaching offering ensures that courses in the nontechnical academic programms follow the specific profiling of TUHH degree courses.

The learning architecture demands and trains independent educational planning as regards the individual development of competences. It also provides orientation knowledge in the form of “profiles”.

The subjects that can be studied in parallel throughout the student’s entire study program - if need be, it can be studied in one to two semesters. In view of the adaptation problems that individuals commonly face in their first semesters after making the transition from school to university and in order to encourage individually planned semesters abroad, there is no obligation to study these subjects in one or two specific semesters during the course of studies.

Teaching and Learning Arrangements

provide for students, separated into B.Sc. and M.Sc., to learn with and from each other across semesters. The challenge of dealing with interdisciplinarity and a variety of stages of learning in courses are part of the learning architecture and are deliberately encouraged in specific courses.

Fields of Teaching

are based on research findings from the academic disciplines cultural studies, social studies, arts, historical studies, communication studies, migration studies and sustainability research, and from engineering didactics. In addition, from the winter semester 2014/15 students on all Bachelor’s courses will have the opportunity to learn about business management and start-ups in a goal-oriented way.

The fields of teaching are augmented by soft skills offers and a foreign language offer. Here, the focus is on encouraging goal-oriented communication skills, e.g. the skills required by outgoing engineers in international and intercultural situations.

The Competence Level

of the courses offered in this area is different as regards the basic training objective in the Bachelor’s and Master’s fields. These differences are reflected in the practical examples used, in content topics that refer to different professional application contexts, and in the higher scientific and theoretical level of abstraction in the B.Sc.

This is also reflected in the different quality of soft skills, which relate to the different team positions and different group leadership functions of Bachelor’s and Master’s graduates in their future working life.

Specialized Competence (Knowledge)

Students can

- explain specialized areas in context of the relevant non-technical disciplines,
- outline basic theories, categories, terminology, models, concepts or artistic techniques in the disciplines represented in the learning area,
- different specialist disciplines relate to their own discipline and differentiate it as well as make connections,
- sketch the basic outlines of how scientific disciplines, paradigms, models, instruments, methods and forms of representation in the specialized sciences are subject to individual and socio-cultural interpretation and historicity,
- Can communicate in a foreign language in a manner appropriate to the subject.

Skills

Professional Competence (Skills)

In selected sub-areas students can

- apply basic and specific methods of the said scientific disciplines,
- aquestion a specific technical phenomena, models, theories from the viewpoint of another, aforementioned specialist discipline,
- to handle simple and advanced questions in aforementioned scientific disciplines in a sucsessful manner,
- justify their decisions on forms of organization and application in practical questions in contexts that go beyond the technical relationship to the subject.

Personal Competence

Social Competence

Personal Competences (Social Skills)

	Students will be able • to learn to collaborate in different manner, • to present and analyze problems in the abovementioned fields in a partner or group situation in a manner appropriate to the addressees, • to express themselves competently, in a culturally appropriate and gender-sensitive manner in the language of the country (as far as this study-focus would be chosen), • to explain nontechnical items to auditorium with technical background knowledge.
Autonomy	**Personal Competences (Self-reliance)** Students are able in selected areas • to reflect on their own profession and professionalism in the context of real-life fields of application • to organize themselves and their own learning processes • to reflect and decide questions in front of a broad education background • to communicate a nontechnical item in a competent way in writen form or verbaly • to organize themselves as an entrepreneurial subject country (as far as this study-focus would be chosen)
Workload in Hours	Depends on choice of courses
Credit points	6

Courses
Information regarding lectures and courses can be found in the corresponding module handbook published separately.

Module M1563: Research Project Computer Science

Courses

Title	Typ	Hrs/wk	CP
Research Project Computer Science (L2353)	Projection Course	8	12

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	Basic knowledge and techniques from the Master courses in the semesters 1 and 2.
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to acquire advanced knowledge in a subfield of Computer Science and can independently acquire deeper knowledge in the field.
Skills	The students are able to formulate the scientific problems to be considered and to work out solutions in an independent manner and to realize them.
Personal Competence	
Social Competence	The students are able to discuss proposals for solutions of scientific problems within the team. They are able to present the results in a clear and well structured manner.
Autonomy	The students can provide a scientific work in a timely manner and document the results in a detailed and well readable form. They are able to actively follow anticipate the presentations of other students such that eventually a scientific discussion comes up.
Workload in Hours	Independent Study Time 248, Study Time in Lecture 112
Credit points	12
Course achievement	None
Examination	Study work
Examination duration and scale	Vortrag
Assignment for the Following Curricula	Computer Science: Core Qualification: Compulsory

Course L2353: Research Project Computer Science

Typ	Projection Course
Hrs/wk	8
CP	12
Workload in Hours	Independent Study Time 248, Study Time in Lecture 112
Lecturer	Prof. Karl-Heinz Zimmermann
Language	DE/EN
Cycle	WiSe
Content	
Literature	

Specialization I. Computer and Software Engineering

Module M0753: Software Verification

Courses

Title	**Typ**	**Hrs/wk**	**CP**
Software Verification (L0629)	Lecture	2	3
Software Verification (L0630)	Recitation Section (small)	2	3

Module Responsible	Prof. Sibylle Schupp
Admission Requirements	None
Recommended Previous Knowledge	• Automata theory and formal languages • Computational logic • Object-oriented programming, algorithms, and data structures • Functional programming or procedural programming • Concurrency
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students apply the major verification techniques in model checking and deductive verification. They explain in formal terms syntax and semantics of the underlying logics, and assess the expressivity of different logics as well as their limitations. They classify formal properties of software systems. They find flaws in formal arguments, arising from modeling artifacts or underspecification.
Skills	Students formulate provable properties of a software system in a formal language. They develop logic-based models that properly abstract from the software under verification and, where necessary, adapt model or property. They construct proofs and property checks by hand or using tools for model checking or deductive verification, and reflect on the scope of the results. Presented with a verification problem in natural language, they select the appropriate verification technique and justify their choice.
Personal Competence	
Social Competence	Students discuss relevant topics in class. They defend their solutions orally. They communicate in English.
Autonomy	Using accompanying on-line material for self study, students can assess their level of knowledge continuously and adjust it appropriately. Working on exercise problems, they receive additional feedback. Within limits, they can set their own learning goals. Upon successful completion, students can identify and precisely formulate new problems in academic or applied research in the field of software verification. Within this field, they can conduct independent studies to acquire the necessary competencies and compile their findings in academic reports. They can devise plans to arrive at new solutions or assess existing ones.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	**Compulsory** Yes / **Bonus** 15 % / **Form** Excercises / **Description**
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computational Science and Engineering: Specialisation I. Computer Science: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Software: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems: Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory

Course L0629: Software Verification	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sibylle Schupp
Language	EN
Cycle	WiSe
Content	• Syntax and semantics of logic-based systems • Deductive verification ◦ Specification ◦ Proof obligations ◦ Program properties ◦ Automated vs. interactive theorem proving • Model checking ◦ Foundations ◦ Property languages ◦ Tool support • Timed automata • Recent developments of verification techniques and applications
Literature	• C. Baier and J-P. Katoen, Principles of Model Checking, MIT Press 2007. • M. Huth and M. Bryan, Logic in Computer Science. Modelling and Reasoning about Systems, 2nd Edition, 2004. • Selected Research Papers

Course L0630: Software Verification	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sibylle Schupp
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M0926: Distributed Algorithms

Courses

Title	Typ	Hrs/wk	CP
Distributed Algorithms (L1071)	Lecture	2	3
Distributed Algorithms (L1072)	Recitation Section (large)	2	3

Module Responsible	Prof. Volker Turau
Admission Requirements	None
Recommended Previous Knowledge	• Algorithms and data structures • Distributed systems • Discrete mathematics • Graph theory
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students know the main abstractions of distributed algorithms (synchronous/asynchronous model, message passing and shared memory model). They are able to describe complexity measures for distributed algorithms (round , message and memory complexity). They explain well known distributed algorithms for important problems such as leader election, mutual exclusion, graph coloring, spanning trees. They know the fundamental techniques used for randomized algorithms.
Skills	Students design their own distributed algorithms and analyze their complexity. They make use of known standard algorithms. They compute the complexity of randomized algorithms.
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	45 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computational Science and Engineering: Specialisation I. Computer Science: Elective Compulsory

Course L1071: Distributed Algorithms

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Volker Turau
Language	DE/EN
Cycle	WiSe
Content	• Leader Election • Colorings & Independent Sets • Tree Algorithms • Minimal Spanning Trees • Randomized Distributed Algorithms • Mutual Exclusion
Literature	1. David Peleg: Distributed Computing - A Locality-Sensitive Approach. SIAM Monograph, 2000 2. Gerard Tel: Introduction to Distributed Algorithms, Cambridge University Press, 2nd edition, 2000 3. Nancy Lynch: Distributed Algorithms. Morgan Kaufmann, 1996 4. Volker Turau: Algorithmische Graphentheorie. Oldenbourg Wissenschaftsverlag, 3. Auflage, 2004.

Course L1072: Distributed Algorithms

Typ	Recitation Section (large)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Volker Turau
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M0942: Software Security

Courses

Title	Typ	Hrs/wk	CP
Software Security (L1103)	Lecture	2	3
Software Security (L1104)	Recitation Section (small)	2	3

Module Responsible	Prof. Dieter Gollmann
Admission Requirements	None
Recommended Previous Knowledge	Familiarity with C/C++, web programming
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can • name the main causes for security vulnerabilities in software • explain current methods for identifying and avoiding security vulnerabilities • explain the fundamental concepts of code-based access control
Skills	Students are capable of • performing a software vulnerability analysis • developing secure code
Personal Competence	
Social Competence	None
Autonomy	Students are capable of acquiring knowledge independently from professional publications, technical standards, and other sources, and are capable of applying newly acquired knowledge to new problems.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	120 minutes
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computational Science and Engineering: Specialisation I. Computer Science: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems: Elective Compulsory

Course L1103: Software Security

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Dieter Gollmann
Language	EN
Cycle	WiSe
Content	• Reliabilty and Software Security • Attacks exploiting character and integer representations • Buffer overruns • Vulnerabilities in memory managemet: double free attacks • Race conditions • SQL injection • Cross-site scripting and cross-site request forgery • Testing for security; taint analysis • Type safe languages • Development proceses for secure software • Code-based access control
Literature	M. Howard, D. LeBlanc: Writing Secure Code, 2nd edition, Microsoft Press (2002) G. Hoglund, G. McGraw: Exploiting Software, Addison-Wesley (2004) L. Gong, G. Ellison, M. Dageforde: Inside Java 2 Platform Security, 2nd edition, Addison-Wesley (2003) B. LaMacchia, S. Lange, M. Lyons, R. Martin, K. T. Price: .NET Framework Security, Addison-Wesley Professional (2002) D. Gollmann: Computer Security, 3rd edition (2011)

Course L1104: Software Security	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Dieter Gollmann
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1400: Design of Dependable Systems

Courses

Title	Typ	Hrs/wk	CP
Designing Dependable Systems (L2000)	Lecture	2	3
Designing Dependable Systems (L2001)	Recitation Section (small)	2	3

Module Responsible	Prof. Görschwin Fey
Admission Requirements	None
Recommended Previous Knowledge	Basic knowledge about data structures and algorithms
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	In the following "dependable" summarizes the concepts Reliability, Availability, Maintainability, Safety and Security. Knowledge about approaches for designing dependable systems, e.g., • Structural solutions like modular redundancy • Algorithmic solutions like handling byzantine faults or checkpointing Knowledge about methods for the analysis of dependable systems
Skills	Ability to implement dependable systems using the above approaches. Ability to analyzs the dependability of systems using the above methods for analysis.
Personal Competence	
Social Competence	Students • discuss relevant topics in class and • present their solutions orally.
Autonomy	Using accompanying material students independently learn in-depth relations between concepts explained in the lecture and additional solution strategies.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6

Course achievement

Compulsory	Bonus	Form	Description
Yes	None	Subject theoretical and practical work	Die Lösung einer Aufgabe ist Zulassungsvoraussetzung für die Prüfung. Die Aufgabe wird in Vorlesung und Übung definiert.

Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computational Science and Engineering: Specialisation I. Computer Science: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Microelectronics and Microsystems: Specialisation Embedded Systems: Elective Compulsory

Course L2000: Designing Dependable Systems	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Görschwin Fey
Language	DE/EN
Cycle	SoSe
Content	Description The term dependability comprises various aspects of a system. These are typically: • Reliability • Availability • Maintainability • Safety • Security This makes dependability a core aspect that has to be considered early in system design, no matter whether software, embedded systems or full scale cyber-physical systems are considered. Contents The module introduces the basic concepts for the design and the analysis of dependable systems. Design examples for getting practical hands-on-experience in dependable design techniques. The module focuses towards embedded systems. The following topics are covered: • Modelling • Fault Tolerance • Design Concepts • Analysis Techniques
Literature	

Course L2001: Designing Dependable Systems	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Görschwin Fey
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M1248: Compilers for Embedded Systems

Courses

Title	Typ	Hrs/wk	CP
Compilers for Embedded Systems (L1692)	Lecture	3	4
Compilers for Embedded Systems (L1693)	Project-/problem-based Learning	1	2

Module Responsible	Prof. Heiko Falk
Admission Requirements	None
Recommended Previous Knowledge	Module "Embedded Systems" C/C++ Programming skills
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The relevance of embedded systems increases from year to year. Within such systems, the amount of software to be executed on embedded processors grows continuously due to its lower costs and higher flexibility. Because of the particular application areas of embedded systems, highly optimized and application-specific processors are deployed. Such highly specialized processors impose high demands on compilers which have to generate code of highest quality. After the successful attendance of this course, the students are able • to illustrate the structure and organization of such compilers, • to distinguish and explain intermediate representations of various abstraction levels, and • to assess optimizations and their underlying problems in all compiler phases. The high demands on compilers for embedded systems make effective code optimizations mandatory. The students learn in particular, • which kinds of optimizations are applicable at the source code level, • how the translation from source code to assembly code is performed, • which kinds of optimizations are applicable at the assembly code level, • how register allocation is performed, and • how memory hierarchies can be exploited effectively. Since compilers for embedded systems often have to optimize for multiple objectives (e.g., average- or worst-case execution time, energy dissipation, code size), the students learn to evaluate the influence of optimizations on these different criteria.
Skills	After successful completion of the course, students shall be able to translate high-level program code into machine code. They will be enabled to assess which kind of code optimization should be applied most effectively at which abstraction level (e.g., source or assembly code) within a compiler. While attending the labs, the students will learn to implement a fully functional compiler including optimizations.
Personal Competence	
Social Competence	Students are able to solve similar problems alone or in a group and to present the results accordingly.
Autonomy	Students are able to acquire new knowledge from specific literature and to associate this knowledge with other classes.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Electrical Engineering: Specialisation Information and Communication Systems: Elective Compulsory Aircraft Systems Engineering: Specialisation Avionic Systems: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory

Course L1692: Compilers for Embedded Systems	
Typ	Lecture
Hrs/wk	3
CP	4
Workload in Hours	Independent Study Time 78, Study Time in Lecture 42
Lecturer	Prof. Heiko Falk
Language	DE/EN
Cycle	SoSe
Content	• Introduction and Motivation • Compilers for Embedded Systems - Requirements and Dependencies • Internal Structure of Compilers • Pre-Pass Optimizations • HIR Optimizations and Transformations • Code Generation • LIR Optimizations and Transformations • Register Allocation • WCET-Aware Compilation • Outlook
Literature	• Peter Marwedel. Embedded System Design - Embedded Systems Foundations of Cyber-Physical Systems. 2 nd Edition, Springer, 2012. • Steven S. Muchnick. Advanced Compiler Design and Implementation. Morgan Kaufmann, 1997. • Andrew W. Appel. Modern compiler implementation in C. Oxford University Press, 1998.

Course L1693: Compilers for Embedded Systems	
Typ	Project-/problem-based Learning
Hrs/wk	1
CP	2
Workload in Hours	Independent Study Time 46, Study Time in Lecture 14
Lecturer	Prof. Heiko Falk
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M1397: Model Checking - Proof Engines and Algorithms

Courses

Title	Typ	Hrs/wk	CP
Model Checking - Proof Engines and Algorithms (L1979)	Lecture	2	3
Model Checking - Proof Engines and Algorithms (L1980)	Recitation Section (small)	2	3

Module Responsible	Prof. Görschwin Fey
Admission Requirements	None
Recommended Previous Knowledge	Basic knowledge about data structures and algorithms
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students know • algorithms and data structures for model checking, • basics of Boolean reasoning engines and • the impact of specification and modelling on the computational effort for model checking.
Skills	Students can • explain and implement algorithms and data structures for model checking, • decide whether a given problem can be solved using Boolean reasoning or model checking, and • implement the respective algorithms.
Personal Competence	
Social Competence	Students • discuss relevant topics in class and • defend their solutions orally.
Autonomy	Using accompanying material students independently learn in-depth relations between concepts explained in the lecture and additional solution strategies.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6

Course achievement	Compulsory	Bonus	Form	Description
	Yes	None	Subject theoretical and practical work	Die Aufgabe wird im Rahmen von Volresung und Prüfung definiert. Die Lösung der Aufgabe ist Zulassungsvoraussetzung für die Prüfung.

Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Software: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems: Elective Compulsory

Course L1979: Model Checking - Proof Engines and Algorithms	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Görschwin Fey
Language	DE/EN
Cycle	SoSe
Content	Correctness is a major concern in embedded systems. Model checking can fully automatically proof formal properties about digital hardware or software. Such properties are given in temporal logic, e.g., to prove "No two orthogonal traffic lights will ever be green." And how do the underlying reasoning algorithms work so effectively in practice despite a computational complexity of NP hardness and beyond? But what are the limitations of model checking? How are the models generated from a given design? The lecture will answer these questions. Open source tools will be used to gather a practical experience. Among other topics, the lecture will consider the following topics: • Modelling digital Hardware, Software, and Cyber Physical Systems • Data structures, decision procedures and proof engines ◦ Binary Decision Diagrams ◦ And-Inverter-Graphs ◦ Boolean Satisfiability ◦ Satisfiability Modulo Theories • Specification Languages ◦ CTL ◦ LTL ◦ System Verilog Assertions • Algorithms for ◦ Reachability Analysis ◦ Symbolic CTL Checking ◦ Bounded LTL-Model Checking ◦ Optimizations, e.g., induction, abstraction • Quality assurance
Literature	Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. 1999. *Model Checking*. MIT Press, Cambridge, MA, USA. A. Biere, A. Biere, M. Heule, H. van Maaren, and T. Walsh. 2009. *Handbook of Satisfiability: Volume 185 Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, The Netherlands, The Netherlands. Selected research papers

Course L1980: Model Checking - Proof Engines and Algorithms	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Görschwin Fey
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M1301: Software Testing

Courses

Title	Typ	Hrs/wk	CP
Software Testing (L1791)	Lecture	2	3
Software Testing (L1792)	Project-/problem-based Learning	2	3

Module Responsible	Prof. Sibylle Schupp
Admission Requirements	None
Recommended Previous Knowledge	• Software Engineering • Higher Programming Languages • Object-Oriented Programming • Algorithms and Data Structures • Experience with (Small) Software Projects • Statistics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students explain the different phases of testing, describe fundamental techniques of different types of testing, and paraphrase the basic principles of the corresponding test process. They give examples of software development scenarios and the corresponding test type and technique. They explain algorithms used for particular testing techniques and describe possible advantages and limitations.
Skills	Students identify the appropriate testing type and technique for a given problem. They adapt and execute respective algorithms to execute a concrete test technique properly. They interpret testing results and execute corresponding steps for proper re-test scenarios. They write and analyze test specifications. They apply bug finding techniques for non-trivial problems.
Personal Competence	
Social Competence	Students discuss relevant topics in class. They defend their solutions orally. They communicate in English.
Autonomy	Students can assess their level of knowledge continuously and adjust it appropriately, based on feedback and on self-guided studies. Within limits, they can own learning goals. Upon successful completion, students can identify and precisely formulate new problems in academic or applied research in the field of testing. Within this field, they can conduct independent studies to acquire the necessary competencies and compile their findings in academic reports. T devise plans to arrive at new solutions or assess existing ones
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Subject theoretical and practical work
Examination duration and scale	Software
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Software: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Software and Signal Processing: Elective Compulsory

Course L1791: Software Testing	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sibylle Schupp
Language	EN
Cycle	SoSe
Content	• Fundamentals of software testing • Model-based testing • Test automation • Criteria-based testing
Literature	• M. Pezze and M. Young, Software Testing and Analysis, John Wiley 2008. • P. Ammann and J. Offutt, "Introduction to Software Testing", 2nd edition 2016. • A. Zeller: "Why Programs Fail: A Guide to Systematic Debugging", 2nd edition 2012.

Course L1792: Software Testing	
Typ	Project-/problem-based Learning
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sibylle Schupp
Language	EN
Cycle	SoSe
Content	• Fundamentals of software testing • Model-based testing • Test automation • Criteria-based testing
Literature	• M. Pezze and M. Young, Software Testing and Analysis, John Wiley 2008. • P. Ammann and J. Offutt, "Introduction to Software Testing", 2nd edition 2015.

Module M0556: Computer Graphics

Courses

Title	**Typ**	**Hrs/wk**	**CP**
Computer Graphics (L0145)	Lecture	2	3
Computer Graphics (L0768)	Recitation Section (small)	2	3

Module Responsible	Prof. Tobias Knopp
Admission Requirements	None
Recommended Previous Knowledge	• Linear Algebra (in particular matrix/vector computation) • Basic programming skills in C/C++
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can explain and describe basic algorithms in 3D computer graphics.
Skills	Students are capable of • implementing a basic 3D rendering pipeline. This consists of projecting simple 3D structures (e.g. cube, spheres) onto a 2D surface using a virtual camera. • apply geometric transformations (e.g. rotation, scaling) in 2D and 3D computer graphics. • using well-known 2D/3D APIs (OpenGL, Cairo) for solving a given problem statement.
Personal Competence	
Social Competence	Students can collaborate in a small team on the realization and validation of a 3D computer graphics pipeline.
Autonomy	• Students are able to solve simple tasks independently with reference to the contents of the lectures and the exercise sets. • Students are able to solve detailed problems independently with the aid of the tutorial’s programming task.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Signal Processing: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Software and Signal Processing: Elective Compulsory

Course L0145: Computer Graphics	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Tobias Knopp
Language	EN
Cycle	SoSe
Content	Computer graphics and animation are leading to an unprecedented visual revolution. The course deals with its technological foundations: • Object-oriented Computer Graphics • Projections and Transformations • Polygonal and Parametric Modelling • Illuminating, Shading, Rendering • Computer Animation Techniques • Kinematics and Dynamics Effects Students will be be working on a series of mini-projects which will eventually evolve into a final project. Learning computer graphics and animation resembles learning a musical instrument. Therefore, doing your projects well and in time is essential for performing well on this course.
Literature	Alan H. Watt: 3D Computer Graphics. Harlow: Pearson (3rd ed., repr., 2009). Dariush Derakhshani: Introducing Autodesk Maya 2014. New York, NY : Wiley (2013).

Course L0768: Computer Graphics	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Tobias Knopp
Language	EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0924: Software for Embedded Systems

Courses

Title	Typ	Hrs/wk	CP
Software for Embdedded Systems (L1069)	Lecture	2	3
Software for Embdedded Systems (L1070)	Recitation Section (small)	3	3

Module Responsible	Prof. Bernd-Christian Renner
Admission Requirements	None
Recommended Previous Knowledge	• Good knowledge and experience in programming language C • Basis knowledge in software engineering • Basic understanding of assembly language
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students know the basic principles and procedures of software engineering for embedded systems. They are able to describe the usage and pros of event based programming using interrupts. They know the components and functions of a concrete microcontroller. The participants explain requirements of real time systems. They know at least three scheduling algorithms for real time operating systems including their pros and cons.
Skills	Students build interrupt-based programs for a concrete microcontroller. They build and use a preemptive scheduler. They use peripheral components (timer, ADC, EEPROM) to realize complex tasks for embedded systems. To interface with external components they utilize serial protocols.
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Electrical Engineering: Specialisation Information and Communication Systems: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Software and Signal Processing: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Software: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Microelectronics and Microsystems: Specialisation Embedded Systems: Elective Compulsory Microelectronics and Microsystems: Specialisation Embedded Systems: Elective Compulsory

Course L1069: Software for Embdedded Systems

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Bernd-Christian Renner
Language	DE/EN
Cycle	SoSe
Content	• General-Purpose Processors • Programming the Atmel AVR • Interrupts • C for Embedded Systems • Standard Single Purpose Processors: Peripherals • Finite-State Machines • Memory • Operating Systems for Embedded Systems • Real-Time Embedded Systems • Boot loader and Power Management
Literature	1. Embedded System Design, F. Vahid and T. Givargis, John Wiley 2. Programming Embedded Systems: With C and Gnu Development Tools, M. Barr and A. Massa, O'Reilly 3. C und C++ für Embedded Systems, F. Bollow, M. Homann, K. Köhn, MITP 4. The Art of Designing Embedded Systems, J. Ganssle, Newnses 5. Mikrocomputertechnik mit Controllern der Atmel AVR-RISC-Familie, G. Schmitt, Oldenbourg 6. Making Embedded Systems: Design Patterns for Great Software, E. White, O'Reilly

Course L1070: Software for Embdedded Systems	
Typ	Recitation Section (small)
Hrs/wk	3
CP	3
Workload in Hours	Independent Study Time 48, Study Time in Lecture 42
Lecturer	Prof. Bernd-Christian Renner
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M1427: Algorithmic Game Theory

Courses

Title	Typ	Hrs/wk	CP
Algorithmic game theory (L2060)	Lecture	2	4
Algorithmic game theory (L2061)	Recitation Section (large)	2	2

Module Responsible	Prof. Matthias Mnich
Admission Requirements	None
Recommended Previous Knowledge	• Mathematics I • Mathematics II • Algorithms and Data Structures
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students can name the basic concepts in algorithmic game theory and mechanism design. They are able to explain them using appropriate examples. • Students can discuss logical connections between these concepts. They are capable of illustrating these connections with the help of examples. • They know game and mechanism design strategies and can reproduce them.
Skills	• Students can model strategic interaction systems of agents with the help of the concepts studied in this course. Moreover, they are capable of analyzing their efficiency and equilibria, by applying established methods. • Students are able to discover and verify further logical connections between the concepts studied in the course. • For a given problem, the students can develop and execute a suitable approach, and are able to critically evaluate the results.
Personal Competence	
Social Competence	• Students are able to work together in teams. They are capable to use mathematics as a common language. • In doing so, they can communicate new concepts according to the needs of their cooperating partners. Moreover, they can design examples to check and deepen the understanding of their peers.
Autonomy	• Students are capable of checking their understanding of complex concepts on their own. They can specify open questions precisely and know where to get help in solving them. • Students have developed sufficient persistence to be able to work for longer periods in a goal-oriented manner on hard problems.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computational Science and Engineering: Specialisation I. Computer Science: Elective Compulsory

Course L2060: Algorithmic game theory	
Typ	Lecture
Hrs/wk	2
CP	4
Workload in Hours	Independent Study Time 92, Study Time in Lecture 28
Lecturer	Prof. Matthias Mnich
Language	DE/EN
Cycle	SoSe
Content	Algorithmic game theory is a topic at the intersection of economics and computation. It deals with analyzing the behavior and interactions of strategic agents, who often try to maximize their incentives. The environment in which those agents interact is referred to as a game. We wish to understand if the agents can reach an "equilibrium", or steady state of the game, in which agents have no incentive to deviate from their chosen strategies. The algorithmic part is to design efficient methods to find equilibria in games, and to make recommendations to the agents so that they can quickly reach a state of personal satisfaction. We will also study mechanism design. In mechanism design, we wish to design markets and auctions and give strategic options to agents, so that they have an incentive to act rationally. We also wish to design the markets and auctions so that they are efficient, in the sense that all goods are cleared and agents do not overpay for the goods which they acquire. Topics: • basic equilibrium concepts (Nash equilibria, correlated equilibria, ...) • strategic actions (best-response dynamics, no-regret dynamics, ...) • auction design (revenue-maximizing auctions, Vickrey auctions) • stable matching theory (preference aggregations, kidney exchanges, ...) • price of anarchy and selfish routing (Braess' paradox, congestion games, ...)
Literature	• T. Roughgarden: Twenty Lectures on Algorithmic Game Theory, Cambridge University Press, 2016. • N. Nisan, T. Roughgarden, E. Tardos, V. Vazirani. Algorithmic Game Theory. Cambridge University Press, 2007.

Course L2061: Algorithmic game theory	
Typ	Recitation Section (large)
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Matthias Mnich
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0839: Traffic Engineering

Courses

Title	Typ	Hrs/wk	CP
Seminar Traffic Engineering (L0902)	Seminar	2	2
Traffic Engineering (L0900)	Lecture	2	2
Traffic Engineering Exercises (L0901)	Recitation Section (small)	1	2

Module Responsible	Prof. Andreas Timm-Giel
Admission Requirements	None
Recommended Previous Knowledge	• Fundamentals of communication or computer networks • Stochastics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to describe methods for planning, optimisation and performance evaluation of communication networks.
Skills	Students are able to solve typical planning and optimisation tasks for communication networks. Furthermore they are able to evaluate the network performance using queuing theory. Students are able to apply independently what they have learned to other and new problems. They can present their results in front of experts and discuss them.
Personal Competence	
Social Competence	
Autonomy	Students are able to acquire the necessary expert knowledge to understand the functionality and performance of new communication networks independently.
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Electrical Engineering: Specialisation Information and Communication Systems: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Networks: Elective Compulsory

Course L0902: Seminar Traffic Engineering

Typ	Seminar
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Andreas Timm-Giel, Dr. Phuong Nga Tran
Language	EN
Cycle	WiSe
Content	Selected applications of methods for planning, optimization, and performance evaluation of communication networks, which have been introduced in the traffic engineering lecture are prepared by the students and presented in a seminar.
Literature	• U. Killat, Entwurf und Analyse von Kommunikationsnetzen, Vieweg + Teubner • further literature announced in the lecture

Course L0900: Traffic Engineering	
Typ	Lecture
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Andreas Timm-Giel, Dr. Phuong Nga Tran
Language	EN
Cycle	WiSe
Content	Network Planning and Optimization • Linear Programming (LP) • Network planning with LP solvers • Planning of communication networks Queueing Theory for Communication Networks • Stochastic processes • Queueing systems • Switches (circuit- and packet switching) • Network of queues
Literature	Literatur: U. Killat, Entwurf und Analyse von Kommunikationsnetzen, Springer Weitere Literatur wird in der Lehrveranstaltung bekanntgegeben / Literature: U. Killat, Entwurf und Analyse von Kommunikationsnetzen, Springer further literature announced in the lecture

Course L0901: Traffic Engineering Exercises	
Typ	Recitation Section (small)
Hrs/wk	1
CP	2
Workload in Hours	Independent Study Time 46, Study Time in Lecture 14
Lecturer	Prof. Andreas Timm-Giel
Language	EN
Cycle	WiSe
Content	Accompanying exercise for the traffic engineering course
Literature	Literatur: U. Killat, Entwurf und Analyse von Kommunikationsnetzen, Springer Weitere Literatur wird in der Lehrveranstaltung bekanntgegeben / Literature: U. Killat, Entwurf und Analyse von Kommunikationsnetzen, Springer further literature announced in the lecture

Module M0910: Advanced System-on-Chip Design (Lab)

Courses

Title	Typ	Hrs/wk	CP
Advanced System-on-Chip Design (L1061)	Project-/problem-based Learning	3	6

Module Responsible	Prof. Heiko Falk
Admission Requirements	None
Recommended Previous Knowledge	Successful completion of the practical FPGA lab of module "Computer Architecture" is a mandatory prerequisite.
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	This module provides in-depth, hands-on experience on advanced concepts of computer architecture. Using the Hardware Description Language VHDL and using reconfigurable FPGA hardware boards, students learn how to design complex computer systems (so-called systems-on-chip, SoCs), that are commonly found in the domain of embedded systems, in actual hardware. Starting with a simple processor architecture, the students learn to how realize instruction-processing of a computer processor according to the principle of pipelining. They implement different styles of cache-based memory hierarchies, examine strategies for dynamic scheduling of machine instructions and for branch prediction, and finally construct a complex MPSoC system (multi-processor system-on-chip) that consists of multiple processor cores that are connected via a shared bus.
Skills	Students will be able to analyze, how highly specific and individual computer systems can be constructed using a library of given standard components. They evaluate the interferences between the physical structure of a computer system and the software executed thereon. This way, they will be enabled to estimate the effects of design decision at the hardware level on the performance of the entire system, to evaluate the whole and complex system and to propose design options to improve a system.
Personal Competence	
Social Competence	Students are able to solve similar problems alone or in a group and to present the results accordingly.
Autonomy	Students are able to acquire new knowledge from specific literature, to transform this knowledge into actual implementations of complex hardware structures, and to associate this knowledge with contents of other classes.
Workload in Hours	Independent Study Time 138, Study Time in Lecture 42
Credit points	6
Course achievement	None
Examination	Subject theoretical and practical work
Examination duration and scale	VHDL Codes and FPGA-based implementations
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Microelectronics and Microsystems: Specialisation Embedded Systems: Elective Compulsory

Course L1061: Advanced System-on-Chip Design

Typ	Project-/problem-based Learning
Hrs/wk	3
CP	6
Workload in Hours	Independent Study Time 138, Study Time in Lecture 42
Lecturer	Prof. Heiko Falk
Language	DE/EN
Cycle	WiSe
Content	• Introduction into fundamental technologies (FPGAs, MIPS single-cycle machine) • Pipelined instruction execution • Cache-based memory hierarchies • Busses and their arbitration • Multi-Processor Systems-on-Chip • Optional: Advanced pipelining concepts (dynamic scheduling, branch prediction)
Literature	• D. Patterson, J. Hennessy. Rechnerorganisation und -entwurf. Elsevier, 2005. • A. Tanenbaum, J. Goodman. Computerarchitektur. Pearson, 2001. • A. Clements. The Principles of Computer Hardware. 3. Auflage, Oxford University Press, 2000.

Module M1742: Operating System Techniques

Courses			
Title	**Typ**	**Hrs/wk**	**CP**
Operating System Techniques (L2815)	Lecture	1	2
Operating System Techniques (L2816)	Project-/problem-based Learning	3	4

Module Responsible	Prof. Christian Dietrich
Admission Requirements	None
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	
Skills	
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory

Course L2815: Operating System Techniques

Typ	Lecture
Hrs/wk	1
CP	2
Workload in Hours	Independent Study Time 46, Study Time in Lecture 14
Lecturer	Prof. Christian Dietrich
Language	DE
Cycle	WiSe
Content	
Literature	

Course L2816: Operating System Techniques

Typ	Project-/problem-based Learning
Hrs/wk	3
CP	4
Workload in Hours	Independent Study Time 78, Study Time in Lecture 42
Lecturer	Prof. Christian Dietrich
Language	DE
Cycle	WiSe
Content	
Literature	

Module M1741: Operating System Construction

Courses

Title	Typ	Hrs/wk	CP
Operating System Construction (L2812)	Lecture	2	3
Operating System Construction (L2814)	Project-/problem-based Learning	2	2
Operating System Construction (L2813)	Recitation Section (large)	1	1

Module Responsible	Prof. Christian Dietrich
Admission Requirements	None
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	
Skills	
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	**Compulsory** No **Bonus** 20 % **Form** Subject theoretical and practical work **Description**
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory Computer Science: Specialisation I. Computer and Software Engineering: Elective Compulsory

Course L2812: Operating System Construction

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Christian Dietrich
Language	DE
Cycle	SoSe
Content	
Literature	

Course L2814: Operating System Construction

Typ	Project-/problem-based Learning
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Christian Dietrich
Language	DE
Cycle	SoSe
Content	
Literature	

Course L2813: Operating System Construction

Typ	Recitation Section (large)
Hrs/wk	1
CP	1
Workload in Hours	Independent Study Time 16, Study Time in Lecture 14
Lecturer	Prof. Christian Dietrich
Language	DE
Cycle	SoSe
Content	
Literature	

Specialization II: Intelligence Engineering

Module M0633: Industrial Process Automation

Courses

Title	**Typ**	**Hrs/wk**	**CP**
Industrial Process Automation (L0344)	Lecture	2	3
Industrial Process Automation (L0345)	Recitation Section (small)	2	3

Module Responsible	Prof. Alexander Schlaefer
Admission Requirements	None
Recommended Previous Knowledge	mathematics and optimization methods principles of automata principles of algorithms and data structures programming skills
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The students can evaluate and assess discrete event systems. They can evaluate properties of processes and explain methods for process analysis. The students can compare methods for process modelling and select an appropriate method for actual problems. They can discuss scheduling methods in the context of actual problems and give a detailed explanation of advantages and disadvantages of different programming methods. The students can relate process automation to methods from robotics and sensor systems as well as to recent topics like 'cyberphysical systems' and 'industry 4.0'.
Skills	The students are able to develop and model processes and evaluate them accordingly. This involves taking into account optimal scheduling, understanding algorithmic complexity, and implementation using PLCs.
Personal Competence	
Social Competence	The students work in teams to solve problems.
Autonomy	The students can reflect their knowledge and document the results of their work.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	**Compulsory** No; **Bonus** 10 %; **Form** Excercises; **Description**
Examination	Written exam
Examination duration and scale	90 minutes
Assignment for the Following Curricula	Bioprocess Engineering: Specialisation A - General Bioprocess Engineering: Elective Compulsory Chemical and Bioprocess Engineering: Specialisation Chemical Process Engineering: Elective Compulsory Chemical and Bioprocess Engineering: Specialisation General Process Engineering: Elective Compulsory Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Control and Power Systems Engineering: Elective Compulsory Aircraft Systems Engineering: Specialisation Cabin Systems: Elective Compulsory International Management and Engineering: Specialisation II. Mechatronics: Elective Compulsory International Management and Engineering: Specialisation II. Product Development and Production: Elective Compulsory Mechanical Engineering and Management: Specialisation Mechatronics: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory Process Engineering: Specialisation Chemical Process Engineering: Elective Compulsory Process Engineering: Specialisation Process Engineering: Elective Compulsory

Course L0344: Industrial Process Automation	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	WiSe
Content	- foundations of problem solving and system modeling, discrete event systems - properties of processes, modeling using automata and Petri-nets - design considerations for processes (mutex, deadlock avoidance, liveness) - optimal scheduling for processes - optimal decisions when planning manufacturing systems, decisions under uncertainty - software design and software architectures for automation, PLCs
Literature	J. Lunze: „Automatisierungstechnik“, Oldenbourg Verlag, 2012 Reisig: Petrinetze: Modellierungstechnik, Analysemethoden, Fallstudien; Vieweg+Teubner 2010 Hrúz, Zhou: Modeling and Control of Discrete-event Dynamic Systems; Springer 2007 Li, Zhou: Deadlock Resolution in Automated Manufacturing Systems, Springer 2009 Pinedo: Planning and Scheduling in Manufacturing and Services, Springer 2009

Course L0345: Industrial Process Automation	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M0550: Digital Image Analysis

Courses

Title	Typ	Hrs/wk	CP
Digital Image Analysis (L0126)	Lecture	4	6

Module Responsible	Prof. Rolf-Rainer Grigat
Admission Requirements	None
Recommended Previous Knowledge	System theory of one-dimensional signals (convolution and correlation, sampling theory, interpolation and decimation, Fourier transform, linear time-invariant systems), linear algebra (Eigenvalue decomposition, SVD), basic stochastics and statistics (expectation values, influence of sample size, correlation and covariance, normal distribution and its parameters), basics of Matlab, basics in optics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can • Describe imaging processes • Depict the physics of sensorics • Explain linear and non-linear filtering of signals • Establish interdisciplinary connections in the subject area and arrange them in their context • Interpret effects of the most important classes of imaging sensors and displays using mathematical methods and physical models.
Skills	Students are able to • Use highly sophisticated methods and procedures of the subject area • Identify problems and develop and implement creative solutions. Students can solve simple arithmetical problems relating to the specification and design of image processing and image analysis systems. Students are able to assess different solution approaches in multidimensional decision-making areas. Students can undertake a prototypical analysis of processes in Matlab.
Personal Competence	
Social Competence	k.A.
Autonomy	Students can solve image analysis tasks independently using the relevant literature.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	60 Minutes, Content of Lecture and materials in StudIP
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Information and Communication Systems: Elective Compulsory Electrical Engineering: Specialisation Medical Technology: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Signal Processing: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Software and Signal Processing: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Microelectronics and Microsystems: Specialisation Communication and Signal Processing: Elective Compulsory Microelectronics and Microsystems: Specialisation Communication and Signal Processing: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Numerics and Computer Science: Elective Compulsory

Course L0126: Digital Image Analysis	
Typ	Lecture
Hrs/wk	4
CP	6
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Lecturer	Prof. Rolf-Rainer Grigat
Language	EN
Cycle	WiSe
Content	• Image representation, definition of images and volume data sets, illumination, radiometry, multispectral imaging, reflectivities, shape from shading • Perception of luminance and color, color spaces and transforms, color matching functions, human visual system, color appearance models • imaging sensors (CMOS, CCD, HDR, X-ray, IR), sensor characterization(EMVA1288), lenses and optics • spatio-temporal sampling (interpolation, decimation, aliasing, leakage, moiré, flicker, apertures) • features (filters, edge detection, morphology, invariance, statistical features, texture) • optical flow (variational methods, quadratic optimization, Euler-Lagrange equations) • segmentation (distance, region growing, cluster analysis, active contours, level sets, energy minimization and graph cuts) • registration (distance and similarity, variational calculus, iterative closest points)
Literature	Bredies/Lorenz, Mathematische Bildverarbeitung, Vieweg, 2011 Wedel/Cremers, Stereo Scene Flow for 3D Motion Analysis, Springer 2011 Handels, Medizinische Bildverarbeitung, Vieweg, 2000 Pratt, Digital Image Processing, Wiley, 2001 Jain, Fundamentals of Digital Image Processing, Prentice Hall, 1989

Module M1336: Soft Computing - Introduction to Machine Learning

Courses

Title	Typ	Hrs/wk	CP
Soft Computing - Introduction to Machine Learning (L1869)	Lecture	4	6

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	Bachelor in Computer Science. Basics in higher mathematics are inevitable, like calculus, linear algebra, graph theory, and optimization.
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to formalize, compute, and analyze belief networks, alignments of sequences, hidden Markov models, phylogenetic tree models, classical regression and clustering methods, neural networks, and fuzzy controllers.
Skills	Students can apply the relevant algorithms and determine their complexity, and they can make use of the statistics language R.
Personal Competence	
Social Competence	Students are able to solve specific problems alone or in a group and to present the results accordingly.
Autonomy	Students are able to acquire new knowledge from newer literature and to associate the acquired knowledge to other fields.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Numerics and Computer Science: Elective Compulsory

Course L1869: Soft Computing - Introduction to Machine Learning

Typ	Lecture
Hrs/wk	4
CP	6
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Lecturer	Prof. Karl-Heinz Zimmermann, Dr. Mehwish Saleemi
Language	DE/EN
Cycle	WiSe
Content	Students are able to formalize, compute, and analyze belief networks, alignments of sequences, hidden Markov models, phylogenetic tree models, neural networks, and fuzzy controllers. In particular, inference and learning in belief networks are important topics that the students should be able to master. Students can apply the relevant algorithms and determine their complexity, and they can make use of the statistics language R.
Literature	1. David Barber, Bayes Reasoning and Machine Learning, Cambridge Univ. Press, Cambridge, 2012. 2. Volker Claus, Stochastische Automaten, Teubner, Stuttgart, 1971. 3. Ernst Klement, Radko Mesiar, Endre Pap, Triangular Norms, Kluwer, Dordrecht, 2000. 4. Timo Koski, John M. Noble, Bayesian Networks, Wiley, New York, 2009. 5. Dimitris Margaritis, Learning Bayesian Network Model Structure from Data, PhD thesis, Carnegie Mellon University, Pittsburgh, 2003. 6. Hidetoshi Nishimori, Statistical Physics of Spin Glasses and Information Processing, Oxford Univ. Press, London, 2001. 7. James R. Norris, Markov Chains, Cambridge Univ. Press, Cambridge, 1996. 8. Maria Rizzo, Statistical Computing with R, Chapman & Hall/CRC, Boca Raton, 2008. 9. Peter Sprites, Clark Glymour, Richard Scheines, Causation, Prediction, and Search, Springer, New York, 1993. 10. Raul Royas, Neural Networks, Springer, Berlin, 1996. 11. Lior Pachter, Bernd Sturmfels, Algebraic Statistics for Computational Biology, Cambridge Univ. Press, Cambridge, 2005. 12. David A. Sprecher, From Algebra to Computational Algorithms, Docent Press, Boston, 2017. 13. Karl-Heinz Zimmermann, Algebraic Statistics, TubDok, Hamburg, 2016.

Module M0629: Intelligent Autonomous Agents and Cognitive Robotics

Courses

Title	Typ	Hrs/wk	CP
Intelligent Autonomous Agents and Cognitive Robotics (L0341)	Lecture	2	4
Intelligent Autonomous Agents and Cognitive Robotics (L0512)	Recitation Section (small)	2	2

Module Responsible	Rainer Marrone
Admission Requirements	None
Recommended Previous Knowledge	Vectors, matrices, Calculus
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can explain the agent abstraction, define intelligence in terms of rational behavior, and give details about agent design (goals, utilities, environments). They can describe the main features of environments. The notion of adversarial agent cooperation can be discussed in terms of decision problems and algorithms for solving these problems. For dealing with uncertainty in real-world scenarios, students can summarize how Bayesian networks can be employed as a knowledge representation and reasoning formalism in static and dynamic settings. In addition, students can define decision making procedures in simple and sequential settings, with and with complete access to the state of the environment. In this context, students can describe techniques for solving (partially observable) Markov decision problems, and they can recall techniques for measuring the value of information. Students can identify techniques for simultaneous localization and mapping, and can explain planning techniques for achieving desired states. Students can explain coordination problems and decision making in a multi-agent setting in term of different types of equilibria, social choice functions, voting protocol, and mechanism design techniques.
Skills	Students can select an appropriate agent architecture for concrete agent application scenarios. For simplified agent application students can derive decision trees and apply basic optimization techniques. For those applications they can also create Bayesian networks/dynamic Bayesian networks and apply bayesian reasoning for simple queries. Students can also name and apply different sampling techniques for simplified agent scenarios. For simple and complex decision making students can compute the best action or policies for concrete settings. In multi-agent situations students will apply techniques for finding different equilibria states,e.g., Nash equilibria. For multi-agent decision making students will apply different voting protocols and compare and explain the results.
Personal Competence	
Social Competence	Students are able to discuss their solutions to problems with others. They communicate in English
Autonomy	Students are able of checking their understanding of complex concepts by solving varaints of concrete problems
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 minutes
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Biomedical Engineering: Specialisation Artificial Organs and Regenerative Medicine: Elective Compulsory Biomedical Engineering: Specialisation Implants and Endoprostheses: Elective Compulsory Biomedical Engineering: Specialisation Medical Technology and Control Theory: Elective Compulsory Biomedical Engineering: Specialisation Management and Business Administration: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Numerics and Computer Science: Elective Compulsory

Course L0341: Intelligent Autonomous Agents and Cognitive Robotics	
Typ	Lecture
Hrs/wk	2
CP	4
Workload in Hours	Independent Study Time 92, Study Time in Lecture 28
Lecturer	Rainer Marrone
Language	EN
Cycle	WiSe
Content	• Definition of agents, rational behavior, goals, utilities, environment types • Adversarial agent cooperation: Agents with complete access to the state(s) of the environment, games, Minimax algorithm, alpha-beta pruning, elements of chance • Uncertainty: Motivation: agents with no direct access to the state(s) of the environment, probabilities, conditional probabilities, product rule, Bayes rule, full joint probability distribution, marginalization, summing out, answering queries, complexity, independence assumptions, naive Bayes, conditional independence assumptions • Bayesian networks: Syntax and semantics of Bayesian networks, answering queries revised (inference by enumeration), typical-case complexity, pragmatics: reasoning from effect (that can be perceived by an agent) to cause (that cannot be directly perceived). • Probabilistic reasoning over time: Environmental state may change even without the agent performing actions, dynamic Bayesian networks, Markov assumption, transition model, sensor model, inference problems: filtering, prediction, smoothing, most-likely explanation, special cases: hidden Markov models, Kalman filters, Exact inferences and approximations • Decision making under uncertainty: Simple decisions: utility theory, multivariate utility functions, dominance, decision networks, value of informatio Complex decisions: sequential decision problems, value iteration, policy iteration, MDPs Decision-theoretic agents: POMDPs, reduction to multidimensional continuous MDPs, dynamic decision networks • Simultaneous Localization and Mapping • Planning • Game theory (Golden Balls: Split or Share) Decisions with multiple agents, Nash equilibrium, Bayes-Nash equilibrium • Social Choice Voting protocols, preferences, paradoxes, Arrow's Theorem, • Mechanism Design Fundamentals, dominant strategy implementation, Revelation Principle, Gibbard-Satterthwaite Impossibility Theorem, Direct mechanisms, incentive compatibility, strategy-proofness, Vickrey-Groves-Clarke mechanisms, expected externality mechanisms, participation constraints, individual rationality, budget balancedness, bilateral trade, Myerson-Satterthwaite Theorem
Literature	1. Artificial Intelligence: A Modern Approach (Third Edition), Stuart Russell, Peter Norvig, Prentice Hall, 2010, Chapters 2-5, 10-11, 13-17 2. Probabilistic Robotics, Thrun, S., Burgard, W., Fox, D. MIT Press 2005 3. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations, Yoav Shoham, Kevin Leyton-Brown, Cambridge University Press, 2009

Course L0512: Intelligent Autonomous Agents and Cognitive Robotics	
Typ	Recitation Section (small)
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Rainer Marrone
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M0630: Robotics and Navigation in Medicine

Courses

Title	Typ	Hrs/wk	CP
Robotics and Navigation in Medicine (L0335)	Lecture	2	3
Robotics and Navigation in Medicine (L0338)	Project Seminar	2	2
Robotics and Navigation in Medicine (L0336)	Recitation Section (small)	1	1

Module Responsible	Prof. Alexander Schlaefer
Admission Requirements	None
Recommended Previous Knowledge	• principles of math (algebra, analysis/calculus) • principles of programming, e.g., in Java or C++ • solid R or Matlab skills
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The students can explain kinematics and tracking systems in clinical contexts and illustrate systems and their components in detail. Systems can be evaluated with respect to collision detection and safety and regulations. Students can assess typical systems regarding design and limitations.
Skills	The students are able to design and evaluate navigation systems and robotic systems for medical applications.
Personal Competence	
Social Competence	The students discuss the results of other groups, provide helpful feedback and can incoorporate feedback into their work.
Autonomy	The students can reflect their knowledge and document the results of their work. They can present the results in an appropriate manner.
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	**Compulsory** / **Bonus** / **Form** / **Description** Yes / 10 % / Written elaboration Yes / 10 % / Presentation
Examination	Written exam
Examination duration and scale	90 minutes
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Medical Technology: Elective Compulsory International Management and Engineering: Specialisation II. Electrical Engineering: Elective Compulsory International Management and Engineering: Specialisation II. Process Engineering and Biotechnology: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Biomedical Engineering: Specialisation Artificial Organs and Regenerative Medicine: Elective Compulsory Biomedical Engineering: Specialisation Implants and Endoprostheses: Elective Compulsory Biomedical Engineering: Specialisation Medical Technology and Control Theory: Elective Compulsory Biomedical Engineering: Specialisation Management and Business Administration: Elective Compulsory Product Development, Materials and Production: Specialisation Product Development: Elective Compulsory Product Development, Materials and Production: Specialisation Production: Elective Compulsory Product Development, Materials and Production: Specialisation Materials: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Bio- and Medical Technology: Elective Compulsory

Course L0335: Robotics and Navigation in Medicine

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	SoSe
Content	- kinematics - calibration - tracking systems - navigation and image guidance - motion compensation The seminar extends and complements the contents of the lecture with respect to recent research results.
Literature	Spong et al.: Robot Modeling and Control, 2005 Troccaz: Medical Robotics, 2012 Further literature will be given in the lecture.

Course L0338: Robotics and Navigation in Medicine	
Typ	Project Seminar
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Course L0336: Robotics and Navigation in Medicine	
Typ	Recitation Section (small)
Hrs/wk	1
CP	1
Workload in Hours	Independent Study Time 16, Study Time in Lecture 14
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0551: Pattern Recognition and Data Compression

Courses

Title	Typ	Hrs/wk	CP
Pattern Recognition and Data Compression (L0128)	Lecture	4	6

Module Responsible	Prof. Rolf-Rainer Grigat
Admission Requirements	None
Recommended Previous Knowledge	Linear algebra (including PCA, unitary transforms), stochastics and statistics, binary arithmetics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can name the basic concepts of pattern recognition and data compression. Students are able to discuss logical connections between the concepts covered in the course and to explain them by means of examples.
Skills	Students can apply statistical methods to classification problems in pattern recognition and to prediction in data compression. On a sound theoretical and methodical basis they can analyze characteristic value assignments and classifications and describe data compression and video signal coding. They are able to use highly sophisticated methods and processes of the subject area. Students are capable of assessing different solution approaches in multidimensional decision-making areas.
Personal Competence	
Social Competence	k.A.
Autonomy	Students are capable of identifying problems independently and of solving them scientifically, using the methods they have learnt.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	60 Minutes, Content of Lecture and materials in StudIP
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Information and Communication Systems: Elective Compulsory Information and Communication Systems: Specialisation Secure and Dependable IT Systems, Focus Software and Signal Processing: Elective Compulsory Information and Communication Systems: Specialisation Communication Systems, Focus Signal Processing: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory International Management and Engineering: Specialisation II. Electrical Engineering: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory

Course L0128: Pattern Recognition and Data Compression	
Typ	Lecture
Hrs/wk	4
CP	6
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Lecturer	Prof. Rolf-Rainer Grigat
Language	EN
Cycle	SoSe
Content	Structure of a pattern recognition system, statistical decision theory, classification based on statistical models, polynomial regression, dimension reduction, multilayer perceptron regression, radial basis functions, support vector machines, unsupervised learning and clustering, algorithm-independent machine learning, mixture models and EM, adaptive basis function models and boosting, Markov random fields Information, entropy, redundancy, mutual information, Markov processes, basic coding schemes (code length, run length coding, prefix-free codes), entropy coding (Huffman, arithmetic coding), dictionary coding (LZ77/Deflate/LZMA2, LZ78/LZW), prediction, DPCM, CALIC, quantization (scalar and vector quantization), transform coding, prediction, decorrelation (DPCM, DCT, hybrid DCT, JPEG, JPEG-LS), motion estimation, subband coding, wavelets, HEVC (H.265,MPEG-H)
Literature	Schürmann: Pattern Classification, Wiley 1996 Murphy, Machine Learning, MIT Press, 2012 Barber, Bayesian Reasoning and Machine Learning, Cambridge, 2012 Duda, Hart, Stork: Pattern Classification, Wiley, 2001 Bishop: Pattern Recognition and Machine Learning, Springer 2006 Salomon, Data Compression, the Complete Reference, Springer, 2000 Sayood, Introduction to Data Compression, Morgan Kaufmann, 2006 Ohm, Multimedia Communication Technology, Springer, 2004 Solari, Digital video and audio compression, McGraw-Hill, 1997 Tekalp, Digital Video Processing, Prentice Hall, 1995

Module M0627: Machine Learning and Data Mining

Courses

Title	Typ	Hrs/wk	CP
Machine Learning and Data Mining (L0340)	Lecture	2	4
Machine Learning and Data Mining (L0510)	Recitation Section (small)	2	2

Module Responsible	NN
Admission Requirements	None
Recommended Previous Knowledge	• Calculus • Stochastics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can explain the difference between instance-based and model-based learning approaches, and they can enumerate basic machine learning technique for each of the two basic approaches, either on the basis of static data, or on the basis of incrementally incoming data . For dealing with uncertainty, students can describe suitable representation formalisms, and they explain how axioms, features, parameters, or structures used in these formalisms can be learned automatically with different algorithms. Students are also able to sketch different clustering techniques. They depict how the performance of learned classifiers can be improved by ensemble learning, and they can summarize how this influences computational learning theory. Algorithms for reinforcement learning can also be explained by students.
Skills	Student derive decision trees and, in turn, propositional rule sets from simple and static data tables and are able to name and explain basic optimization techniques. They present and apply the basic idea of first-order inductive leaning. Students apply the BME, MAP, ML, and EM algorithms for learning parameters of Bayesian networks and compare the different algorithms. They also know how to carry out Gaussian mixture learning. They can contrast kNN classifiers, neural networks, and support vector machines, and name their basic application areas and algorithmic properties. Students can describe basic clustering techniques and explain the basic components of those techniques. Students compare related machine learning techniques, e.g., k-means clustering and nearest neighbor classification. They can distinguish various ensemble learning techniques and compare the different goals of those techniques.
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 minutes
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory International Management and Engineering: Specialisation II. Information Technology: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory

Course L0340: Machine Learning and Data Mining	
Typ	Lecture
Hrs/wk	2
CP	4
Workload in Hours	Independent Study Time 92, Study Time in Lecture 28
Lecturer	Rainer Marrone
Language	EN
Cycle	SoSe
Content	- Decision trees - First-order inductive learning - Incremental learning: Version spaces - Uncertainty - Bayesian networks - Learning parameters of Bayesian networks BME, MAP, ML, EM algorithm - Learning structures of Bayesian networks - Gaussian Mixture Models - kNN classifier, neural network classifier, support vector machine (SVM) classifier - Clustering Distance measures, k-means clustering, nearest neighbor clustering - Kernel Density Estimation - Ensemble Learning - Reinforcement Learning - Computational Learning Theory
Literature	1. Artificial Intelligence: A Modern Approach (Third Edition), Stuart Russel, Peter Norvig, Prentice Hall, 2010, Chapters 13, 14, 18-21 2. Machine Learning: A Probabilistic Perspective, Kevin Murphy, MIT Press 2012

Course L0510: Machine Learning and Data Mining	
Typ	Recitation Section (small)
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Rainer Marrone
Language	EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0623: Intelligent Systems in Medicine

Courses

Title	Typ	Hrs/wk	CP
Intelligent Systems in Medicine (L0331)	Lecture	2	3
Intelligent Systems in Medicine (L0334)	Project Seminar	2	2
Intelligent Systems in Medicine (L0333)	Recitation Section (small)	1	1

Module Responsible	Prof. Alexander Schlaefer
Admission Requirements	None
Recommended Previous Knowledge	• principles of math (algebra, analysis/calculus) • principles of stochastics • principles of programming, Java/C++ and R/Matlab • advanced programming skills
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The students are able to analyze and solve clinical treatment planning and decision support problems using methods for search, optimization, and planning. They are able to explain methods for classification and their respective advantages and disadvantages in clinical contexts. The students can compare different methods for representing medical knowledge. They can evaluate methods in the context of clinical data and explain challenges due to the clinical nature of the data and its acquisition and due to privacy and safety requirements.
Skills	The students can give reasons for selecting and adapting methods for classification, regression, and prediction. They can assess the methods based on actual patient data and evaluate the implemented methods.
Personal Competence	
Social Competence	The students discuss the results of other groups, provide helpful feedback and can incoorporate feedback into their work.
Autonomy	The students can reflect their knowledge and document the results of their work. They can present the results in an appropriate manner.
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	**Compulsory** / **Bonus** / **Form** / **Description** Yes / 10 % / Presentation Yes / 10 % / Written elaboration
Examination	Written exam
Examination duration and scale	90 minutes
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Medical Technology: Elective Compulsory Interdisciplinary Mathematics: Specialisation Computational Methods in Biomedical Imaging: Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Biomedical Engineering: Specialisation Artificial Organs and Regenerative Medicine: Elective Compulsory Biomedical Engineering: Specialisation Implants and Endoprostheses: Elective Compulsory Biomedical Engineering: Specialisation Medical Technology and Control Theory: Elective Compulsory Biomedical Engineering: Specialisation Management and Business Administration: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Bio- and Medical Technology: Elective Compulsory

Course L0331: Intelligent Systems in Medicine

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	WiSe
Content	- methods for search, optimization, planning, classification, regression and prediction in a clinical context - representation of medical knowledge - understanding challenges due to clinical and patient related data and data acquisition The students will work in groups to apply the methods introduced during the lecture using problem based learning.
Literature	Russel & Norvig: Artificial Intelligence: a Modern Approach, 2012 Berner: Clinical Decision Support Systems: Theory and Practice, 2007 Greenes: Clinical Decision Support: The Road Ahead, 2007 Further literature will be given in the lecture

Course L0334: Intelligent Systems in Medicine	
Typ	Project Seminar
Hrs/wk	2
CP	2
Workload in Hours	Independent Study Time 32, Study Time in Lecture 28
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Course L0333: Intelligent Systems in Medicine	
Typ	Recitation Section (small)
Hrs/wk	1
CP	1
Workload in Hours	Independent Study Time 16, Study Time in Lecture 14
Lecturer	Prof. Alexander Schlaefer
Language	EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1302: Applied Humanoid Robotics

Courses

Title	Typ	Hrs/wk	CP
Applied Humanoid Robotics (L1794)	Project-/problem-based Learning	6	6

Module Responsible	Patrick Göttsch
Admission Requirements	None
Recommended Previous Knowledge	• Object oriented programming; algorithms and data structures • Introduction to control systems • Control systems theory and design • Mechanics
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students can explain humanoid robots. • Students can explain the basic concepts, relationships and methods of forward- and inverse kinematics • Students learn to apply basic control concepts for different tasks in humanoid robotics.
Skills	• Students can implement models for humanoid robotic systems in Matlab and C++, and use these models for robot motion or other tasks. • They are capable of using models in Matlab for simulation and testing these models if necessary with C++ code on the real robot system. • They are capable of selecting methods for solving abstract problems, for which no standard methods are available, and apply it successfully.
Personal Competence	
Social Competence	• Students can develop joint solutions in mixed teams and present these. • They can provide appropriate feedback to others, and constructively handle feedback on their own results
Autonomy	• Students are able to obtain required information from provided literature sources, and to put in into the context of the lecture. • They can independently define tasks and apply the appropriate means to solve them.
Workload in Hours	Independent Study Time 96, Study Time in Lecture 84
Credit points	6
Course achievement	None
Examination	Written elaboration
Examination duration and scale	5-10 pages
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Control and Power Systems Engineering: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Bio- and Medical Technology: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory

Course L1794: Applied Humanoid Robotics

Typ	Project-/problem-based Learning
Hrs/wk	6
CP	6
Workload in Hours	Independent Study Time 96, Study Time in Lecture 84
Lecturer	Patrick Göttsch
Language	DE/EN
Cycle	WiSe/SoSe
Content	• Fundamentals of kinematics • Static and dynamic stability of humanoid robotic systems • Combination of different software environments (Matlab, C++, etc.) • Introduction to the necessary software frameworks • Team project • Presentation and Demonstration of intermediate and final results
Literature	• B. Siciliano, O. Khatib. "Handbook of Robotics. Part A: Robotics Foundations", Springer (2008)

Module M1249: Medical Imaging

Courses			
Title	**Typ**	**Hrs/wk**	**CP**
Medical Imaging (L1694)	Lecture	2	3
Medical Imaging (L1695)	Recitation Section (small)	2	3

Module Responsible	Prof. Tobias Knopp
Admission Requirements	None
Recommended Previous Knowledge	Basic knowledge in linear algebra, numerics, and signal processing
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	After successful completion of the module, students are able to describe reconstruction methods for different tomographic imaging modalities such as computed tomography and magnetic resonance imaging. They know the necessary basics from the fields of signal processing and inverse problems and are familiar with both analytical and iterative image reconstruction methods. The students have a deepened knowledge of the imaging operators of computed tomography and magnetic resonance imaging.
Skills	The students are able to implement reconstruction methods and test them using tomographic measurement data. They can visualize the reconstructed images and evaluate the quality of their data and results. In addition, students can estimate the temporal complexity of imaging algorithms.
Personal Competence	
Social Competence	Students can work on complex problems both independently and in teams. They can exchange ideas with each other and use their individual strengths to solve the problem.
Autonomy	Students are able to independently investigate a complex problem and assess which competencies are required to solve it.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Computer Science: Specialisation II: Intelligence Engineering: Elective Compulsory Electrical Engineering: Specialisation Medical Technology: Elective Compulsory Interdisciplinary Mathematics: Specialisation Computational Methods in Biomedical Imaging: Compulsory Microelectronics and Microsystems: Specialisation Communication and Signal Processing: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Bio- and Medical Technology: Elective Compulsory

Course L1694: Medical Imaging

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Tobias Knopp
Language	DE/EN
Cycle	WiSe
Content	• Overview about different imaging methods • Signal processing • Inverse problems • Computed tomography • Magnetic resonance imaging • Compressed Sensing • Magnetic particle imaging
Literature	**Bildgebende Verfahren in der Medizin**; O. Dössel; Springer, Berlin, 2000 **Bildgebende Systeme für die medizinische Diagnostik**; H. Morneburg (Hrsg.); Publicis MCD, München, 1995 **Introduction to the Mathematics of Medical Imaging**; C. L.Epstein; Siam, Philadelphia, 2008 **Medical Image Processing, Reconstruction and Restoration**; J. Jan; Taylor and Francis, Boca Raton, 2006 **Principles of Magnetic Resonance Imaging**; Z.-P. Liang and P. C. Lauterbur; IEEE Press, New York, 1999

Course L1695: Medical Imaging	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Tobias Knopp
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Specialization III. Mathematics

Module M0667: Algorithmic Algebra

Courses

Title	Typ	Hrs/wk	CP
Algorithmic Algebra (L0422)	Lecture	3	5
Algorithmic Algebra (L0423)	Recitation Section (small)	1	1

Module Responsible	Dr. Prashant Batra
Admission Requirements	None
Recommended Previous Knowledge	Mathe I-III (Real analysis,computing in Vector spaces , principle of complete induction) Diskrete Mathematik I (gropus, rings, ideals, fields; euclidean algorithm)
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students can discuss logical connections between the following concepts and explain them by means of examples: Smith normal form, Chinese remainder theorem, grid point sets, integer solution of inequality systems.
Skills	Students are able to access independently further logical connections between the concepts with which they have become familiar and are able to verify them. Students are able to develop a suitable solution approach to given problems, to pursue it and to evaluate the results critically, such as in solving multivariate equation systems and in grid point theory.
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory

Course L0422: Algorithmic Algebra	
Typ	Lecture
Hrs/wk	3
CP	5
Workload in Hours	Independent Study Time 108, Study Time in Lecture 42
Lecturer	Dr. Prashant Batra
Language	DE
Cycle	WiSe
Content	Extended euclidean algorithm, solution of the Bezout-equation Division with remainder (over rings) fast arithmetic algorithms (conversion, fast multiplications) discrete Fourier-transformation over rings Computation with modular remainders, solving of remainder systems (chinese remainder theorem), solvability of integer linear systems over the integers linearization of polynomial equations-- matrix approach Sylvester-matrix, elimination elimination in rings, elimination of many variables Buchberger algorithm, Gröbner basis Minkowskis Lattice Point theorem and integer-valued optimization LLL-algorithm for construction of 'short' lattice vectors in polynomial time
Literature	von zur Gathen, Joachim; Gerhard, Jürgen Modern computer algebra. 3rd ed. (English) Zbl 1277.68002 Cambridge: Cambridge University Press (ISBN 978-1-107-03903-2/hbk; 978-1-139-85606-5/ebook). Yap, Chee Keng Fundamental problems of algorithmic algebra. (English) Zbl 0999.68261 Oxford: Oxford University Press. xvi, 511 p. $ 87.00 (2000). Free download for students from author's website: http://cs.nyu.edu/yap/book/berlin/ Cox, David; Little, John; O'Shea, Donal Ideals, varieties, and algorithms. An introduction to computational algebraic geometry and commutative algebra. 3rd ed. (English) Zbl 1118.13001 Undergraduate Texts in Mathematics. New York, NY: Springer (ISBN 978-0-387-35650-1/hbk; 978-0-387-35651-8/ebook). xv, 551 p. eBook: http://dx.doi.org/10.1007/978-0-387-35651-8 Concrete abstract algebra : from numbers to Gröbner bases / Niels **Lauritzen** **Verfasser:** **Lauritzen,** Niels **Ausgabe:** Reprinted with corr. **Erschienen:** Cambridge [u.a.] : Cambridge Univ. Press, 2006 **Umfang:** XIV, 240 S. : graph. Darst. **Anmerkung:** Includes bibliographical references and index **ISBN:** 0-521-82679-9, 978-0-521-82679-2 (hbk.) : GBP 55.00 0-521-53410-0, 978-0-521-53410-9 (pbk.) : USD 39.99 Koepf, Wolfram Computer algebra. An algorithmic oriented introduction. (Computeralgebra. Eine algorithmisch orientierte Einführung.) (German) Zbl 1161.68881 Berlin: Springer (ISBN 3-540-29894-0/pbk). xiii, 515 p. springer eBook: http://dx.doi.org/10.1007/3-540-29895-9 Kaplan, Michael Computer algebra. (Computeralgebra.) (German) Zbl 1093.68148 Berlin: Springer (ISBN 3-540-21379-1/pbk). xii, 391 p. springer eBook: http://dx.doi.org/10.1007/b137968

Course L0423: Algorithmic Algebra	
Typ	Recitation Section (small)
Hrs/wk	1
CP	1
Workload in Hours	Independent Study Time 16, Study Time in Lecture 14
Lecturer	Dr. Prashant Batra
Language	DE
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1428: Linear and Nonlinear Optimization

Courses

Title	Typ	Hrs/wk	CP
Linear and Nonlinear Optimization (L2062)	Lecture	4	4
Linear and Nonlinear Optimization (L2063)	Recitation Section (large)	1	2

Module Responsible	Prof. Matthias Mnich
Admission Requirements	None
Recommended Previous Knowledge	• Discrete Algebraic Structures • Mathematics I • Graph Theory and Optimization
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students can name the basic concepts in linear and non-linear optimization. They are able to explain them using appropriate examples. • Students can discuss logical connections between these concepts. They are capable of illustrating these connections with the help of examples. • They know proof strategies and can reproduce them.
Skills	• Students can model problems in linear and non-linear optimization with the help of the concepts studied in this course. Moreover, they are capable of solving them by applying established methods. • Students are able to discover and verify further logical connections between the concepts studied in the course. • For a given problem, the students can develop and execute a suitable approach, and are able to critically evaluate the results.
Personal Competence	
Social Competence	• Students are able to work together in teams. They are capable to use mathematics as a common language. • In doing so, they can communicate new concepts according to the needs of their cooperating partners. Moreover, they can design examples to check and deepen the understanding of their peers.
Autonomy	• Students are capable of checking their understanding of complex concepts on their own. They can specify open questions precisely and know where to get help in solving them. • Students have developed sufficient persistence to be able to work for longer periods in a goal-oriented manner on hard problems.
Workload in Hours	Independent Study Time 110, Study Time in Lecture 70
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Computational Science and Engineering: Specialisation III. Mathematics: Elective Compulsory

Course L2062: Linear and Nonlinear Optimization

Typ	Lecture
Hrs/wk	4
CP	4
Workload in Hours	Independent Study Time 64, Study Time in Lecture 56
Lecturer	Prof. Matthias Mnich
Language	DE/EN
Cycle	WiSe
Content	• Modelling linear programming problems • Graphical method • Algebraic background • Convexity • Polyhedral theory • Simplex method • Degeneracy and convergence • duality • interior-point methods • quadratic optimization • integer linear programming
Literature	• A. Schrijver: Combinatorial Optimization: Polyhedra and Efficiency. Springer, 2003 • B. Korte and T. Vygen: Combinatorial Optimization: Theory and Algorithms. Springer, 2018 • T. Cormen, Ch. Leiserson, R. Rivest, C. Stein: Introduction to Algorithms. MIT Press, 2013

Course L2063: Linear and Nonlinear Optimization	
Typ	Recitation Section (large)
Hrs/wk	1
CP	2
Workload in Hours	Independent Study Time 46, Study Time in Lecture 14
Lecturer	Prof. Matthias Mnich
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M0716: Hierarchical Algorithms

Courses

Title	Typ	Hrs/wk	CP
Hierarchical Algorithms (L0585)	Lecture	2	3
Hierarchical Algorithms (L0586)	Recitation Section (small)	2	3

Module Responsible	Prof. Sabine Le Borne
Admission Requirements	None
Recommended Previous Knowledge	• Mathematics I, II, III for Engineering students (german or english) or Analysis & Linear Algebra I + II as well as Analysis III for Technomathematicians • Programming experience in C
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to • name representatives of hierarchical algorithms and list their characteristics, • explain construction techniques for hierarchical algorithms, • discuss aspects regarding the efficient implementation of hierarchical algorithms.
Skills	Students are able to • implement the hierarchical algorithms discussed in the lecture, • analyse the storage and computational complexities of the algorithms, • adapt algorithms to problem settings of various applications and thus develop problem adapted variants.
Personal Competence	
Social Competence	Students are able to • work together in heterogeneously composed teams (i.e., teams from different study programs and background knowledge), explain theoretical foundations and support each other with practical aspects regarding the implementation of algorithms.
Autonomy	Students are capable • to assess whether the supporting theoretical and practical excercises are better solved individually or in a team, • to work on complex problems over an extended period of time, • to assess their individual progess and, if necessary, to ask questions and seek help.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	20 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Mathematical Modelling in Engineering: Theory, Numerics, Applications: Specialisation II. Modelling and Simulation of Complex Systems (TUHH): Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Numerics and Computer Science: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Simulation Technology: Elective Compulsory

Course L0585: Hierarchical Algorithms

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sabine Le Borne
Language	DE/EN
Cycle	WiSe
Content	• Low rank matrices • Separable expansions • Hierarchical matrix partitions • Hierarchical matrices • Formatted matrix operations • Applications • Additional topics (e.g. H2 matrices, matrix functions, tensor products)
Literature	W. Hackbusch: Hierarchische Matrizen: Algorithmen und Analysis

Course L0586: Hierarchical Algorithms	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sabine Le Borne
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1337: Curves, Cryptosystems and Quantum Computing

Courses

Title	Typ	Hrs/wk	CP
Curves, Cryptosystems and Quantum Computing (L1870)	Lecture	4	6

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	Higher algebra, linear algebra, and mathematical analysis.
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The students understand the basic theory of elliptic curves, classical cryptosysteme, basic methods of cryptanalysis, cryptography of elliptic curves, quantum computing and the post-quantum computing scenario.
Skills	The students are in the position to apply the group law of elliptic curves, to find out if a curve is non-singular, to sketch cryptographic algorithms that make use of elliptic curves and to specify quantum algorithms.
Personal Competence	
Social Competence	Students are able to solve specific problems alone or in a group and to present the results accordingly.
Autonomy	Students are able to acquire new knowledge from specific standard books and to associate the acquired knowledge to other classes.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory

Course L1870: Curves, Cryptosystems and Quantum Computing

Typ	Lecture
Hrs/wk	4
CP	6
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Lecturer	Prof. Karl-Heinz Zimmermann
Language	DE/EN
Cycle	SoSe
Content	
Literature	

Module M1310: Discrete Differential Geometry

Courses

Title	Typ	Hrs/wk	CP
Discrete Differential Geometry (L1808)	Lecture	4	6

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	Linear Algebra, Multivariate Calculus
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	These lectures are on geometrical aspects of the solutions of differential equations and their treatment on the computer. The required basics from linear algebra and analysis are reviewed at the beginning. Applications are to curved surfaces in space, to mechanics and mechatronics, to different types of field equations, and to the tranfer of mathematical constructions to data types, compiler functions, programming languages, and special compute circuits. - basic prerequisites from linear algebra, tensors, exterior algebra, Clifford algebras - basic prerequisites from coordinate-free analysis, vector fields and differential forms, integration, discretization - local differential geometry: connections, symplectic geometry and Hamiltonian systems, Riemannian geometry, discretization - global differential geometry: manifolds, Lie groups, fiber bundles, random processes, space and time
Skills	
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory

Course L1808: Discrete Differential Geometry

Typ	Lecture
Hrs/wk	4
CP	6
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Lecturer	Prof. Georg Friedrich Mayer-Lindenberg
Language	DE/EN
Cycle	SoSe
Content	These lectures deal with geometric aspects of differential equations and with their treatment on the computer. The prerequisites from linear algebra and analysis are reviewed at the beginning. Applications are to curved surfaces, to classical mechanics and mechatronics, to various field equations, to computer graphics and to transferring mathematical constructions to data types, compiler functions, programming languages, and special hardware. Keywords: Basics from linear algebra, tensors, exterior algebra, Clifford algebras, tuple types Basics of coordinate-free analysis, vector fields and differential forms, integration, discrete exterior calculus Local differential geometry: connections, symplectic geometry, Riemannian geometry, discrete mechanics and connections Global differential geometry: manifolds, Lie groups, fibre bundles, Fourier decompositions, random processes, space and time
Literature	Agricola, Friedrich, Vektoranalysis, Vieweg/Teubner 2010 A.C. Da Silva, Lectures on Symplectic Geometry, Springer L.N. Math. 1764 J. Snygg, Differential Geometry using Clifford's Algebra, Birkhäuser 2010 T. Frankel, The Geometry of Physics, Cambridge U. P. 2012 M.Desbrun et al., Discrete exterior calculus, arXiv:math/0508341v2 J.Marsden et al., Discrete Mechanics and Variational Integrators, Acta numerica. 2001

Module M1405: Randomised Algorithms and Random Graphs

Courses

Title	Typ	Hrs/wk	CP
Randomised Algorithms and Random Graphs (L2010)	Lecture	2	3
Randomised Algorithms and Random Graphs (L2011)	Recitation Section (large)	2	3

Module Responsible	Prof. Anusch Taraz
Admission Requirements	None
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students can describe basic concepts in the area of Randomized Algorithms and Random Graphs such as random walks, tail bounds, fingerprinting and algebraic techniques, first and second moment methods, and various random graph models. They are able to explain them using appropriate examples. • Students can discuss logical connections between these concepts. They are capable of illustrating these connections with the help of examples. • They know proof strategies and can apply them.
Skills	• Students can model problems with the help of the concepts studied in this course. Moreover, they are capable of solving them by applying established methods. • Students are able to explore and verify further logical connections between the concepts studied in the course. • For a given problem, the students can develop and execute a suitable technique, and are able to critically evaluate the results.
Personal Competence	
Social Competence	• Students are able to work together in teams. They are capable to establish a common language. • In doing so, they can communicate new concepts according to the needs of their cooperating partners. Moreover, they can design examples to check and deepen the understanding of their peers.
Autonomy	• Students are capable of checking their understanding of complex concepts on their own. They can specify open questions precisely and know where to get help in solving them. • Students have developed sufficient persistence to be able to work for longer periods in a goal-oriented manner on hard problems.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	30 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Computational Science and Engineering: Specialisation III. Mathematics: Elective Compulsory Mathematical Modelling in Engineering: Theory, Numerics, Applications: Specialisation I. Numerics (TUHH): Elective Compulsory

Course L2010: Randomised Algorithms and Random Graphs	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Anusch Taraz, Prof. Volker Turau
Language	DE/EN
Cycle	SoSe
Content	Randomized Algorithms: • introduction and recalling basic tools from probability • randomized search • random walks • text search with fingerprinting • parallel and distributed algorithms • online algorithms Random Graphs: • typical properties • first and second moment method • tail bounds • thresholds and phase transitions • probabilistic method • models for complex networks
Literature	• Motwani, Raghavan: Randomized Algorithms • Worsch: Randomisierte Algorithmen • Dietzfelbinger: Randomisierte Algorithmen • Bollobas: Random Graphs • Alon, Spencer: The Probabilistic Method • Frieze, Karonski: Random Graphs • van der Hofstad: Random Graphs and Complex Networks

Course L2011: Randomised Algorithms and Random Graphs	
Typ	Recitation Section (large)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Anusch Taraz, Prof. Volker Turau
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0711: Numerical Mathematics II

Courses

Title	Typ	Hrs/wk	CP
Numerical Mathematics II (L0568)	Lecture	2	3
Numerical Mathematics II (L0569)	Recitation Section (small)	2	3

Module Responsible	Prof. Sabine Le Borne
Admission Requirements	None
Recommended Previous Knowledge	• Numerical Mathematics I • MATLAB knowledge
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to • name advanced numerical methods for interpolation, integration, linear least squares problems, eigenvalue problems, nonlinear root finding problems and explain their core ideas, • repeat convergence statements for the numerical methods, • sketch convergence proofs, • explain practical aspects of numerical methods concerning runtime and storage needs explain aspects regarding the practical implementation of numerical methods with respect to computational and storage complexity. •
Skills	Students are able to • implement, apply and compare advanced numerical methods in MATLAB, • justify the convergence behaviour of numerical methods with respect to the problem and solution algorithm and to transfer it to related problems, • for a given problem, develop a suitable solution approach, if necessary through composition of several algorithms, to execute this approach and to critically evaluate the results
Personal Competence	
Social Competence	Students are able to • work together in heterogeneously composed teams (i.e., teams from different study programs and background knowledge), explain theoretical foundations and support each other with practical aspects regarding the implementation of algorithms.
Autonomy	Students are capable • to assess whether the supporting theoretical and practical excercises are better solved individually or in a team, • to assess their individual progess and, if necessary, to ask questions and seek help.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Computational Science and Engineering: Specialisation III. Mathematics: Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Technical Complementary Course: Elective Compulsory Theoretical Mechanical Engineering: Core Qualification: Elective Compulsory

Course L0568: Numerical Mathematics II	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sabine Le Borne, Dr. Jens-Peter Zemke
Language	DE/EN
Cycle	SoSe
Content	1. Error and stability: Notions and estimates 2. Interpolation: Rational and trigonometric interpolation 3. Quadrature: Gaussian quadrature, orthogonal polynomials 4. Linear systems: Perturbation theory of decompositions, structured matrices 5. Eigenvalue problems: LR-, QD-, QR-Algorithmus 6. Krylov space methods: Arnoldi-, Lanczos methods
Literature	• Stoer/Bulirsch: Numerische Mathematik 1, Springer • Dahmen, Reusken: Numerik für Ingenieure und Naturwissenschaftler, Springer

Course L0569: Numerical Mathematics II	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Sabine Le Borne, Dr. Jens-Peter Zemke
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0714: Numerical Treatment of Ordinary Differential Equations

Courses

Title	Typ	Hrs/wk	CP
Numerical Treatment of Ordinary Differential Equations (L0576)	Lecture	2	3
Numerical Treatment of Ordinary Differential Equations (L0582)	Recitation Section (small)	2	3

Module Responsible	Prof. Daniel Ruprecht
Admission Requirements	None
Recommended Previous Knowledge	• Mathematik I, II, III für Ingenieurstudierende (deutsch oder englisch) oder Analysis & Lineare Algebra I + II sowie Analysis III für Technomathematiker • Basic MATLAB knowledge
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to • list numerical methods for the solution of ordinary differential equations and explain their core ideas, • repeat convergence statements for the treated numerical methods (including the prerequisites tied to the underlying problem), • explain aspects regarding the practical execution of a method. • select the appropriate numerical method for concrete problems, implement the numerical algorithms efficiently and interpret the numerical results
Skills	Students are able to • implement (MATLAB), apply and compare numerical methods for the solution of ordinary differential equations, • to justify the convergence behaviour of numerical methods with respect to the posed problem and selected algorithm, • for a given problem, develop a suitable solution approach, if necessary by the composition of several algorithms, to execute this approach and to critically evaluate the results.
Personal Competence	
Social Competence	Students are able to • work together in heterogeneously composed teams (i.e., teams from different study programs and background knowledge), explain theoretical foundations and support each other with practical aspects regarding the implementation of algorithms.
Autonomy	Students are capable • to assess whether the supporting theoretical and practical excercises are better solved individually or in a team, • to assess their individual progress and, if necessary, to ask questions and seek help.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Written exam
Examination duration and scale	90 min
Assignment for the Following Curricula	Bioprocess Engineering: Specialisation A - General Bioprocess Engineering: Elective Compulsory Chemical and Bioprocess Engineering: Specialisation Chemical Process Engineering: Elective Compulsory Chemical and Bioprocess Engineering: Specialisation General Process Engineering: Elective Compulsory Computer Science: Specialisation III. Mathematics: Elective Compulsory Electrical Engineering: Specialisation Control and Power Systems Engineering: Elective Compulsory Energy Systems: Core Qualification: Elective Compulsory Aircraft Systems Engineering: Specialisation Aircraft Systems: Elective Compulsory Mathematical Modelling in Engineering: Theory, Numerics, Applications: Specialisation l. Numerics (TUHH): Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Core Qualification: Compulsory Process Engineering: Specialisation Chemical Process Engineering: Elective Compulsory Process Engineering: Specialisation Process Engineering: Elective Compulsory

Course L0576: Numerical Treatment of Ordinary Differential Equations	
Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Daniel Ruprecht
Language	DE/EN
Cycle	SoSe
Content	Numerical methods for Initial Value Problems • single step methods • multistep methods • stiff problems • differential algebraic equations (DAE) of index 1 Numerical methods for Boundary Value Problems • multiple shooting method • difference methods • variational methods
Literature	• E. Hairer, S. Noersett, G. Wanner: Solving Ordinary Differential Equations I: Nonstiff Problems • E. Hairer, G. Wanner: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems

Course L0582: Numerical Treatment of Ordinary Differential Equations	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Daniel Ruprecht
Language	DE/EN
Cycle	SoSe
Content	See interlocking course
Literature	See interlocking course

Module M0881: Mathematical Image Processing

Courses

Title	Typ	Hrs/wk	CP
Mathematical Image Processing (L0991)	Lecture	3	4
Mathematical Image Processing (L0992)	Recitation Section (small)	1	2

Module Responsible	Prof. Marko Lindner
Admission Requirements	None
Recommended Previous Knowledge	• Analysis: partial derivatives, gradient, directional derivative • Linear Algebra: eigenvalues, least squares solution of a linear system
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to • characterize and compare diffusion equations • explain elementary methods of image processing • explain methods of image segmentation and registration • sketch and interrelate basic concepts of functional analysis
Skills	Students are able to • implement and apply elementary methods of image processing • explain and apply modern methods of image processing
Personal Competence	
Social Competence	Students are able to work together in heterogeneously composed teams (i.e., teams from different study programs and background knowledge) and to explain theoretical foundations.
Autonomy	• Students are capable of checking their understanding of complex concepts on their own. They can specify open questions precisely and know where to get help in solving them. • Students have developed sufficient persistence to be able to work for longer periods in a goal-oriented manner on hard problems.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	20 min
Assignment for the Following Curricula	Bioprocess Engineering: Specialisation A - General Bioprocess Engineering: Elective Compulsory Computer Science: Specialisation III. Mathematics: Elective Compulsory Computational Science and Engineering: Specialisation III. Mathematics: Elective Compulsory Interdisciplinary Mathematics: Specialisation Computational Methods in Biomedical Imaging: Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Mechatronics: Specialisation System Design: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory Process Engineering: Specialisation Process Engineering: Elective Compulsory

Course L0991: Mathematical Image Processing

Typ	Lecture
Hrs/wk	3
CP	4
Workload in Hours	Independent Study Time 78, Study Time in Lecture 42
Lecturer	Prof. Marko Lindner
Language	DE/EN
Cycle	WiSe
Content	• basic methods of image processing • smoothing filters • the diffusion / heat equation • variational formulations in image processing • edge detection • de-convolution • inpainting • image segmentation • image registration
Literature	Bredies/Lorenz: Mathematische Bildverarbeitung

Course L0992: Mathematical Image Processing	
Typ	Recitation Section (small)
Hrs/wk	1
CP	2
Workload in Hours	Independent Study Time 46, Study Time in Lecture 14
Lecturer	Prof. Marko Lindner
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1552: Mathematics of Neural Networks

Courses

Title	Typ	Hrs/wk	CP
Mathematics of Neural Networks (L2322)	Lecture	2	3
Mathematics of Neural Networks (L2323)	Recitation Section (small)	2	3

Module Responsible	Dr. Jens-Peter Zemke
Admission Requirements	None
Recommended Previous Knowledge	1. Mathematics I-III 2. Numerical Mathematics 1/ Numerics 3. Programming skills, preferably in Python
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	Students are able to name, state and classify state-of-the-art neural networks and their corresponding mathematical basics. They can assess the difficulties of different neural networks.
Skills	Students are able to implement, understand, and, tailored to the field of application, apply neural networks.
Personal Competence	
Social Competence	Students can • develop and document joint solutions in small teams; • form groups to further develop the ideas and transfer them to other areas of applicability; • form a team to develop, build, and advance a software library.
Autonomy	Students are able to • correctly assess the time and effort of self-defined work; • assess whether the supporting theoretical and practical excercises are better solved individually or in a team; • define test problems for testing and expanding the methods; • assess their individual progess and, if necessary, to ask questions and seek help.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Computational Science and Engineering: Specialisation III. Mathematics: Elective Compulsory Mechatronics: Specialisation Intelligent Systems and Robotics: Elective Compulsory Mechatronics: Technical Complementary Course: Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Robotics and Computer Science: Elective Compulsory

Course L2322: Mathematics of Neural Networks

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Dr. Jens-Peter Zemke
Language	DE/EN
Cycle	WiSe
Content	1. Basics: analogy; layout of neural nets, universal approximation, NP-completeness 2. Feedforward nets: backpropagation, variants of Stochastic Gradients 3. Deep Learning: problems and solution strategies 4. Deep Belief Networks: energy based models, Contrastive Divergence 5. CNN: idea, layout, FFT and Winograds algorithms, implementation details 6. RNN: idea, dynamical systems, training, LSTM 7. ResNN: idea, relation to neural ODEs 8. Standard libraries: Tensorflow, Keras, PyTorch 9. Recent trends
Literature	1. Skript 2. Online-Werke: ◦ http://neuralnetworksanddeeplearning.com/ ◦ https://www.deeplearningbook.org/

Course L2323: Mathematics of Neural Networks	
Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Dr. Jens-Peter Zemke
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Module M1020: Numerics of Partial Differential Equations

Courses

Title	Typ	Hrs/wk	CP
Numerics of Partial Differential Equations (L1247)	Lecture	2	3
Numerics of Partial Differential Equations (L1248)	Recitation Section (small)	2	3

Module Responsible	Prof. Daniel Ruprecht
Admission Requirements	None
Recommended Previous Knowledge	• Mathematik I - IV (for Engineering Students) **or** Analysis & Linear Algebra I + II for Technomathematicians • Numerical mathematics 1 • Numerical treatment of ordinary differential equations
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• Students can classify partial differential equations according to the three basic types. • For each type, students know suitable numerical approaches. • Students know the theoretical convergence results for these approaches.
Skills	Students are capable to formulate solution strategies for given problems involving partial differential equations, to comment on theoretical properties concerning convergence and to implement and test these methods in practice.
Personal Competence	
Social Competence	Students are able to work together in heterogeneously composed teams (i.e., teams from different study programs and background knowledge) and to explain theoretical foundations.
Autonomy	• Students are capable of checking their understanding of complex concepts on their own. They can specify open questions precisely and know where to get help in solving them. • Students have developed sufficient persistence to be able to work for longer periods in a goal-oriented manner on hard problems.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Oral exam
Examination duration and scale	25 min
Assignment for the Following Curricula	Computer Science: Specialisation III. Mathematics: Elective Compulsory Technomathematics: Specialisation I. Mathematics: Elective Compulsory Theoretical Mechanical Engineering: Specialisation Simulation Technology: Elective Compulsory

Course L1247: Numerics of Partial Differential Equations

Typ	Lecture
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Daniel Ruprecht
Language	DE/EN
Cycle	WiSe
Content	Elementary Theory and Numerics of PDEs • types of PDEs • well posed problems • finite differences • finite elements • finite volumes • applications
Literature	Dietrich Braess: Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie, Berlin u.a., Springer 2007 Susanne Brenner, Ridgway Scott: The Mathematical Theory of Finite Element Methods, Springer, 2008 Peter Deuflhard, Martin Weiser: Numerische Mathematik 3

Course L1248: Numerics of Partial Differential Equations

Typ	Recitation Section (small)
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Daniel Ruprecht
Language	DE/EN
Cycle	WiSe
Content	See interlocking course
Literature	See interlocking course

Specialization IV. Subject Specific Focus

Module M1565: Technical Complementary Course I for CSMS

Courses			
Title	Typ	Hrs/wk	CP

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	
Skills	
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Depends on choice of courses
Credit points	6
Assignment for the Following Curricula	Computer Science: Specialisation IV. Subject Specific Focus: Elective Compulsory

Module M1566: Technical Complementary Course II for CSMS

Courses			
Title	**Typ**	**Hrs/wk**	**CP**

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	
Skills	
Personal Competence	
Social Competence	
Autonomy	
Workload in Hours	Depends on choice of courses
Credit points	6
Assignment for the Following Curricula	Computer Science: Specialisation IV. Subject Specific Focus: Elective Compulsory

Module M1564: Advanced Seminars Computer Science

Courses

Title	Typ	Hrs/wk	CP
Advanced Seminar Computer Science I (L2352)	Seminar	2	3
Introductory Seminar Computer Science II (L2429)	Seminar	2	3

Module Responsible	Prof. Karl-Heinz Zimmermann
Admission Requirements	None
Recommended Previous Knowledge	Basic knowledge of Computer Science and Mathematics at the Master's level.
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	The students are able to • explicate a specific topic in the field of Computer Science, • describe complex issues, • present different views and evaluate in a critical way.
Skills	The students are able to • familiarize in a specific topic of Computer Science in limited time, • realize a literature survey on the specific topic and cite in a correct way, • elaborate a presentation and give a lecture to a selected audience, • sum up the presentation in 10-15 lines, • answer questions in the final discussion.
Personal Competence	
Social Competence	The students are able to • elaborate and introduce a topic for a certain audience, • discuss the topic, content and structure of the presentation with the instructor, • discuss certain aspects with the audience, and • as the lecturer listen and respond to questions from the audience.
Autonomy	The students are able to • define the task in question in an autonomous way, • develop the necessary knowledge, • use appropriate work equipment, and • guided by an instructor critically check the working status.
Workload in Hours	Independent Study Time 124, Study Time in Lecture 56
Credit points	6
Course achievement	None
Examination	Presentation
Examination duration and scale	x
Assignment for the Following Curricula	Computer Science: Specialisation IV. Subject Specific Focus: Elective Compulsory

Course L2352: Advanced Seminar Computer Science I

Typ	Seminar
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Karl-Heinz Zimmermann
Language	DE/EN
Cycle	WiSe/SoSe
Content	
Literature	

Course L2429: Introductory Seminar Computer Science II

Typ	Seminar
Hrs/wk	2
CP	3
Workload in Hours	Independent Study Time 62, Study Time in Lecture 28
Lecturer	Prof. Karl-Heinz Zimmermann
Language	DE/EN
Cycle	WiSe/SoSe
Content	
Literature	

Thesis

Module M-002: Master Thesis

Courses			
Title	**Typ**	**Hrs/wk**	**CP**

Module Responsible	Professoren der TUHH
Admission Requirements	• According to General Regulations §21 (1): At least 60 credit points have to be achieved in study programme. The examinations board decides on exceptions.
Recommended Previous Knowledge	
Educational Objectives	After taking part successfully, students have reached the following learning results
Professional Competence	
Knowledge	• The students can use specialized knowledge (facts, theories, and methods) of their subject competently on specialized issues. • The students can explain in depth the relevant approaches and terminologies in one or more areas of their subject, describing current developments and taking up a critical position on them. • The students can place a research task in their subject area in its context and describe and critically assess the state of research.
Skills	The students are able: • To select, apply and, if necessary, develop further methods that are suitable for solving the specialized problem in question. • To apply knowledge they have acquired and methods they have learnt in the course of their studies to complex and/or incompletely defined problems in a solution-oriented way. • To develop new scientific findings in their subject area and subject them to a critical assessment.
Personal Competence	
Social Competence	Students can • Both in writing and orally outline a scientific issue for an expert audience accurately, understandably and in a structured way. • Deal with issues competently in an expert discussion and answer them in a manner that is appropriate to the addressees while upholding their own assessments and viewpoints convincingly.
Autonomy	Students are able: • To structure a project of their own in work packages and to work them off accordingly. • To work their way in depth into a largely unknown subject and to access the information required for them to do so. • To apply the techniques of scientific work comprehensively in research of their own.
Workload in Hours	Independent Study Time 900, Study Time in Lecture 0
Credit points	30
Course achievement	None
Examination	Thesis
Examination duration and scale	According to General Regulations
Assignment for the Following Curricula	Civil Engineering: Thesis: Compulsory Bioprocess Engineering: Thesis: Compulsory Chemical and Bioprocess Engineering: Thesis: Compulsory Computer Science: Thesis: Compulsory Electrical Engineering: Thesis: Compulsory Energy and Environmental Engineering: Thesis: Compulsory Energy Systems: Thesis: Compulsory Environmental Engineering: Thesis: Compulsory Aircraft Systems Engineering: Thesis: Compulsory Global Innovation Management: Thesis: Compulsory Computational Science and Engineering: Thesis: Compulsory Information and Communication Systems: Thesis: Compulsory Interdisciplinary Mathematics: Thesis: Compulsory International Management and Engineering: Thesis: Compulsory Joint European Master in Environmental Studies - Cities and Sustainability: Thesis: Compulsory Logistics, Infrastructure and Mobility: Thesis: Compulsory Materials Science: Thesis: Compulsory Mechanical Engineering and Management: Thesis: Compulsory Mechatronics: Thesis: Compulsory Biomedical Engineering: Thesis: Compulsory Microelectronics and Microsystems: Thesis: Compulsory Product Development, Materials and Production: Thesis: Compulsory Renewable Energies: Thesis: Compulsory

	Naval Architecture and Ocean Engineering: Thesis: Compulsory Ship and Offshore Technology: Thesis: Compulsory Teilstudiengang Lehramt Metalltechnik: Thesis: Compulsory Theoretical Mechanical Engineering: Thesis: Compulsory Process Engineering: Thesis: Compulsory Water and Environmental Engineering: Thesis: Compulsory Certification in Engineering & Advisory in Aviation: Thesis: Compulsory