Universität Konstanz

Computer and Information Science M.Sc.

Module guide

October 2020

Contact person:

Dr. Martin Brunner
Department of Computer and Information Science
Phone: +49 7531 88-4437
E-mail: martin.brunner@uni-konstanz.de

– informatik.uni.kn

Content

Master of Science in Computer and Information Science

The international Master study programme is aimed at graduates who have completed bachelor‘s degrees in Computer Science or Information Engineering as well as suitably qualified people from other fields who have an above-average university degree in a related subject.

In terms of its direction of focus, this master study programme belongs to the more research-oriented degree programmes. Teaching is conducted by lecturers whose experiences is particularly based on current research. Students are integrated into ongoing research projects at an early stage of their studies as part of the required project and expand on this knowledge later during the master’s thesis.

This also means that the department’s research focus is reflected in the content of the master’s programme. Notwithstanding individual specialisation, graduates will have competences in the area of methods and systems of visualisation, analysis, exploration, and processing of large amounts of information. This enables them to work in the large information spaces that characterise our information society from a use-oriented perspective. They are able to search, filter and organise data and prepare, summarise and present it as information. For this aim, the programme communicates the methods, applications and evaluations of the systems in automatic information processing based on the principles of computer science and mathematics.

The master courses are held in English. The aim here is to prepare graduates for careers with internationally-operating companies, projects and further postgraduate studies.

Learning outcomes of the study programme

The master's degree programme in Computer and Information Science provides opportunities for specialising in a field of computer and information science, especially in methods and systems of visualisation, analysis, exploration, and processing of large volumes of data. Graduates gain in-depth expertise in their field of specialisation.

In addition to the subject-specific competences also interdisciplinary competences and key qualifications are strengthened. Graduates are able to apply scientific methods of computer and information science to solving complex practical and scientific problems.

The master's degree documents that the graduate has the qualifications required for doing advanced research in the field of computer and information science. Therefore the programme qualifies graduates for responsible tasks in all kinds of companies and organisations nationally as well as internationally, and to apply for admission for doctoral studies.

Structure of the study programme

The standard period of study for the master's programme is four semesters, amounting to 120 ECTS in total.

The study programme is divided into an area of basic modules, an area of advanced modules, an area of supplementary courses as well as a final part of studies. The basic and advanced modules are required elective modules in which students acquire the core competencies of the subject and can choose between several modules.

In the area of basic modules, students complete a minimum of four and a maximum of nine modules à 6 ECTS credits from the Department of Computer and Information Science. Basic modules are, as a rule, lectures with exercises, e.g., Data mining: Basic concepts, Data visualization: Basic concepts, Computer graphics, Interactive systems or Big data management and analysis.

In the area of advanced modules, students complete modules amounting to 36 to 66 ECTS-credits. This area comprises a seminar (3 ECTS credits), a master's project (9 ECTS credits) as well as 24 to 54 ECTS credits from modules from the curriculum of the Department of Computer and Information Science. These modules usually are lectures with tutorials or directed studies, e.g. Data mining: Advanced topics, Data visualization: Advanced topics, Global illumination methods, Interaction in mixed reality spaces, Compiler construction or Randomized algorithms.

The area of supplementary courses comprises cross-disciplinary courses or courses from other departments amounting to a total of 18 ECTS credits. A subject-related transferable skills course in academic writing is mandatory if a course in which the same or comparable skills can be acquired was not already completed during the bachelor’s programme.

The area of basic modules, the area of advanced modules as well as the area of supplementary courses should be completed in the first three semesters of the master’s programme. The forth semester comprises the final part of studies, consisting of the master’s thesis and corresponding colloquium (thesis defence) amounting to 30 ECTS credits.

The overall grade is based on the results of the course-related performance assessments, the master's thesis and the colloquium on the master's thesis.

Students can specialize within the master's programme, e.g. in Data science, Visual computing, Interactive systems, Software systems engineering or Algorithmics. In addition, an individual specialization can be chosen. A specialization can be stated in the examination certificate, if the master’s seminar, the master's project and the master’s thesis are thematically related to the area of specialization and at least three advanced modules from that area of specialization were successfully completed.

Structure of the master's programme

The stated number of weekly teaching hours (SWS) is not binding. It merely indicates the extent of on-campus studies to be expected.

Semester	Areas	SWS (weekly teaching hours)	ECTS credits
	Area of basic modules		
1-3	A minimum of four and maximum of nine modules à 6 ECTS credits from the Department of Computer and Information Science* e.g. Data Visualization: Basic Concepts, Data Mining: Basic Concepts, Computer Graphics, Interactive Systems	16-36	24-54
	Area of advanced modules		
1-3	Master modules from the Department of Computer and Information Science (lectures with exercises and directed studies worth 6-9 ECTS credits each)	16-36	24-54
2 or 3	Seminar	2	3
2 or 3	Master's project	-	9
	Area of supplementary courses (maximum of 18 ECTS credits)		
1-3	Courses from other departments*; courses related to computer science can only be recognized after the Examination Board (*StPA*) (§ 5) has given its approval.	0-12	0-18
1-3	Transferable skills courses offered by the Department of Computer and Information Science, the Centre for Transferable Skills (SQ), the Language Institute and the International Office*; A subject-related transferable skills course in academic writing is mandatory if the student did not complete such a course during the bachelor's programme	0-4	0-6
	Final part of studies		
4	Master's thesis and colloquium (thesis defence)	-	30
Total		**54**	**120**

*It must be made sure that these courses were not already credited in the previous bachelor's programme.

Module directory

This chapter contains detailed descriptions of the individual modules. As both the basic and the advanced modules are required elective courses and the selection of modules offered may change from semester to semester, this section lists those modules that are offered relatively regularly, usually once a year.

The department's Study Commission is responsible for quality assurance of the offered modules. It ensures that a sufficient number of modules from all areas of research are available each semester. The current range of courses is published in the electronic course catalogue in good time before the beginning of each semester.

The modules in this module guide are not assigned to any specialisation, since there is a relatively large degree of freedom of choice in the choice of specialisation and the modules belonging to it. An overview of the specified specializations and the associated modules can be found on the department's homepage at

https://www.informatik.uni-konstanz.de/study/master-of-science/specializations/.

Description of basic modules

A minimum of four and a maximum of nine basic modules à 6 ECTS credits each has to be completed. Basic modules are generally lectures with exercises. In the following list, basic modules which are offered relatively regularly are included. An overview of the basic modules offered in the respective semester can be found in ZEuS.

Module Advanced Algorithmic Concepts

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Knowledge of advanced algorithms concepts and data structures and the ability to evaluate and compare their performance and suitability.
Contents	In this course, we will consider selected tasks and methods in algorithmics. While examplary problems arise from various domains, an emphasis is put on graph algorithms. We will discuss topics such as matroids, network flow algorithms, approximation and fixed parameter algorithms.
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Credits für diese Einheit	6
Coursework and performance assessment	Graded weekly assignments (50% threshold) and oral examination, active participation in the tutorial including at least two presentations of your solutions.
Prerequisites	Basic knowledge of algorithms and data structures.
Language	English
Frequency of offer	at irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Algorithms in Bioinformatics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students gain knowledge of fundamental algorithms in bioinformatics.
Contents	Algorithms are central to the evaluation of biological data. This course deals with essential bioinformatics algorithms, including for the study of sequences, evolutionary relationships of organisms, protein structures and biological networks. Knowledge from the compulsory lecture "Algorithms and Data Structures" is required. The course has a scope of 4 SWS: 2 SWS lecture and 2 SWS exercise. Subjects: - Terms and concepts from biology - Analysis of sequences (sequence comparison, sequence motifs, etc.) - Phylogenetic trees - RNA structure prediction - Analysis of biological networks / graphs - DNA computing - Different algorithmic concepts and their application in bioinformatics
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Credits für diese Einheit	6
Coursework and performance assessment	Depending on the number of participants: oral exam (20 min) or written exam (60 min).
Prerequisites	Knowledge from the course "algorithms and data structure" is required. Special knowledge from biology is not required, the first lecture will give a short introduction to important concepts and terms from biology.
Language	English
Frequency of offer	at irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Agile UX Design

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	- Students know the differences between usability and user experience and why both design goals are important qualities of modern user interfaces. - Students know the main characteristics of agile software development processes and how they are applied in UX design. - Students know how UX design activities are imbedded in agile software development lifecycles (e.g. user stories in sprints). - Students know the different steps of a typical UX lifecycle model (analyze, design, prototype, evaluate) and how they are related to each other. - Students know and can apply the different methods and techniques used in a typical UX lifecycle (e.g. contextual inquiry, user role models, task models, sketching, prototyping, inspection methods). - Students know how to structure and manage a typical UX design project. - Students can apply all the knowledge of this lecture designing user interfaces in medium size projects in small teams (e.g. app development for different application domains).
Contents	This course is about an agile process for UX design, where UX is short for user experience, which includes usability, usefulness, emotional impact, and meaningfulness. We present a process, method, and technique approach to overall UX design based on the UX lifecycle. The basic UX lifecycle activities we will cover include Understand Needs, Design Solutions, Prototype Candidates, and Evaluate UX. It is a goal of this course to help students realize that UX design is an ongoing process throughout the full product or system life cycle, and that developing the UX design is not something to be done at the last minute, when the "rest of the system" is finished. The process, principles, and guidelines are universal and applicable to any kind of design that involves interaction between humans and non-human systems in the broadest sense. The material of this course applies not just to GUIs and the Web but to all kinds of interaction styles and devices, including ATMs, refrigerators, elevator buttons, road signs, ubiquitous computing, embedded computing, and everyday things.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	In-class exercises, exam and team project
Prerequisites	No official coursework is required as a prerequisite, but students should have substantial experience with computers, especially their interactive use, and an intense interest in making them easier to use. Some knowledge of software engineering fundamentals, gained either from coursework or from practice, would be useful but is not essential. We expect most students in this course to be from Computer Science and Human Factors Engineering. However, we do welcome the diversity of others (e.g., art and design, psychology, communications studies, technical writing, music, etc.). With a little effort in learning the technical aspects of processes,

	people in other majors will be able to participate and learn in this course.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Algorithm Engineering

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students gain a basic understanding of how algorithms with theoretical runtime guarantees behave in practice on real-world data. They know about ways to model and analyse hardware-dependent runtime aspects as memory hierarchies and parallelization. Students are able to select among a set of different algorithms the one best suitable for a certain application, and know methods to demonstrate the efficiency of such algorithms.
Contents	Theoretical (worst case) runtimes of algorithms often do not explain their behavior in practice sufficiently. The input data, the hardware architecture, hidden constants and many other factors influence empirical runtimes. Algorithm Engineering tries to bridge this gap between theory and practice. The goal is to design algorithms which work well on real-world data, are simple and easy to implement and take the used hardware into account (e.g. provide good data locality). In the lecture, we will discuss AE approaches for several important applications as sorting, graph partitioning/clustering, route planning and NP-hard problems as TSP, Independent Set and Colouring. We will compare different algorithms on real-world benchmarks, try to identify 'hard inputs' and discuss performance measures which can be compared in a hardware-independent way.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study (exercises and project work).
Coursework and performance assessment	Written exam
Prerequisites	Modules Computer Science 1 + 2
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Big Data Management and Analysis

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The students know and understand the basic concepts for dealing with very large data sets and are able to apply them in small projects.
Contents	The term “big data“ is often used to describe vast collections of semi-structured data in the range of tera- or even petabytes. Companies like Google and Amazon illustrate that mining and analyzing such collections yields the potential for completely new applications. The lecture provides an overview of motivations to analyze big data and introduces techniques needed in the process. This includes introductions to scripting languages, NoSQL databases and Map/Reduce systems which are accompanied by practical exercises. Content overview: - Streaming Algorithms - Memory Hierarchies - Parallel Computations - Storage Area Networks and Distributed File Systems For implementations the students will learn and use the language Python.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study (exercises and project work).
Coursework and performance assessment	Students have to pass 50% of the weekly theoretical and practical assignments and a written exam at the end of the semester.
Prerequisites	Module Computer Science 1 (Konzepte der Informatik + Programmierkurs 1) Module Systems 2 (Datenbanksysteme)
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Computer Graphics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Graduates of the course have a basic understanding of the design of graphical interactive systems and their implementation with OpenGL and shaders. They have in-depth knowledge of the rasterization pipeline and can apply and deploy them in different contexts.
Contents	The lecture provides an introduction to interactive computer graphics with OpenGL and shaders. The students learn about the path from the input data (geometric description of objects) to the pixel of the output image: - Preparation of data (transformation, projection, clipping) - Rasterization (scanline rendering, depth buffering) - Shading methods (Gourand shading, Phong shading) - Local vs. global illumination methods - Raytracing, radiosity, and image-based rendering - Texture mapping - Applications such as computer games, simulators, etc.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study, programming
Coursework and performance assessment	Exam: written exam (60 min). Passing the tutorial is the admission requirement for the final written exam. The final grade is the grade of the written exam.
Prerequisites	Corresponding to the modules Informatics 1 and Systems 1: elementary programming knowledge. Knowledge of C++ or a different object oriented programming language and the willingness to learn C++.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Concurrent, Distributed and Cloud Computing

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	This course will enable participants to design concurrent, distributed and cloud systems as well as to assess the architectures of existing systems. It will therefore enable them to make adequate decisions in projects aimed at designing and operating these types of systems.
Contents	Concurrent, Distributed and Cloud Computing addresses fundamental problems of distributed computing and various levels of abstraction. Cloud computing covers synchronization and resource protection in concurrent programming. Distributed Computing strives to achieve transparency of the distributed nature of systems by defining suitable abstractions, algorithms and middleware semantics. Cloud computing reflects the current trend to relocate distributed computing and storage resources in a fully transparent fashion into the realms of cloud service providers. This course will present foundational algorithms and architectures that have been developed in support of these computing paradigms. It will also address modeling and verification techniques used in the design of concurrent, distributed and cloud computing systems.
Course type / weekly teaching hours	Directed Studies (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Graded weekly assignments (50% threshold) and oral examination, active participation in the tutorial including at least two presentations of your solutions.
Prerequisites	-
Language	English
Frequency of offer	at irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Data Mining: Basic Concepts

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students are taught elementary theoretical knowledge and get first practical experience in the data analysis domain. They obtain the ability to assess requirements and parameters for the application of fundamental analysis algorithms. Beyond that, students will practically apply and assess the results in an autonomous way.

Contents	- Data preprocessing
	- Basic data mining algorithms and methods:
	    o Classification
	    o Clustering
	- Association Rules

Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Written exam or oral exam (depends on the number of students) and successful attendance of the tutorial (at least 50% of reachable points). The final grading only reflects the performance in the exam.
Prerequisites	Modules Computer Science 1 and Mathematics 2
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Data Visualization: Basic Concepts

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students understand the principles of Information Visualization: - Students are able to preprocess, analyze and visualize large amounts of unknown data. - Students are able to analyze existing Information Visualization systems with respect to effectiveness and expressiveness, and systematically design systems for new application areas.
Contents	"Data Visualization: Basic Concepts“ gives an introduction to the field of Data Visualization. In particular, it covers foundations, relevant aspects of human perception, visualization design principles, and some basic visualization techniques for different data types (e.g., multi-dimensional, hierarchical, and spatial).
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	Depending on the number of participants, oral exam (of 30 minutes duration), or written exam (of 120 minutes duration). Eligibility to take part in the exam requires students to achieve at least 50% of the points from the exercise/tutorial program. The final grade corresponds to the grade of the exam.
Prerequisites	The lectures Database Systems, Module Computer Science 1 and 2 are mandatory. Basic programming skills and basic knowledge of databases and query languages are mandatory.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Decision Procedures for Software Verification

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Participants will learn the foundations of SMT technologies. This will enable them to assess SMT technology in its usefulness for solving various problems in academia and industrial practice, and to use this technology in own related projects.
Contents	Software is logic at work, and hence formal mathematical logic is the calculus of software engineering. In the verification of properties of software systems special decidable fragments of first order logic have received special interest and lots of tools have been developed under the label "Satisfiability Modulo Theories" (SMT). Companies like Google and Microsoft, for instance, use this technology to analyze their software eco systems for security vulnerabilities. In this directed studies course we will provide a brief introduction into propositional and first order logic. We will then discuss more SMT specific logics and decision procedures, including DPLL(T) for SAT solving, the Nelson-Oppen combination procedure, uninterpreted functions, linear real and integer arithmetic, and quantifier elimination. In addition, we will consider logics and procedures relevant to software verification problems, including the treatment of arrays and pointers, and provide application examples.
Course type / weekly teaching hours	Directed Studies (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Final examination - oral. A grade bonus can be earned during the tutorials.
Prerequisites	Undergraduate understanding in Algorithms, Programming, Discrete Structures (in particular formal logic), Theory of Computing.
Language	English
Frequency of offer	at irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Digital Signal Processing

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students master the basic methods in digital signal processing and have an in-depth understanding of the concept and methods of continuous-time and discrete-time Fourier transformations. They are able to analyze time-discrete signals and systems and have experience in the practice of digital signal processing using MATLAB or other programming languages.
Contents	This introductory course on digital signal processing covers - discrete-time signals and systems - digital filters - disrete-time Fourier analysis - sampling and resampling of signals - Fourier analysis using the fast Fourier transform Exercises include programming using MATLAB.
Course type / weekly teaching hours	Lecture (3 SWS) and exercise (1 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	The credit requirements include solving homework problems and a written exam.
Prerequisites	- basic math courses offered in our bachelor programs - algorithms and data structures - introduction to computer science including programming
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Document Analysis: Computational Methods

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The students are able to name the problems of document analysis and the basic algorithms for their solution. The students are able to transfer the algorithms to a specific document analysis problem and are able to assess their solution.
Contents	- Natural Language Processing - Information Retrieval - Structure Analysis and Information Extraction - Text Data Mining - Text Visualization
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study and practical assignments.
Coursework and performance assessment	Working on the exercise is a prerequisite for admission to examination. Examination: depending on the number of participants oral (individual exam of 20 minutes) or written (written exam 90 minutes). Grade: The grade results from the grade of the examination.
Prerequisites	A successful previous participation in the course Data Mining is mandatory. The necessary prerequisites for participating in this lecture may be tested as part of an initial exam. Programming skills in Java are necessary to solve the practical exercises.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Illustrative Computer Graphics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Understanding, analyzing and evaluating of algorithms of Non-Photorealistic Computer Graphics. The course focusses on 2D-, 2.5D- and 3D methods to create images aside from standard photo realistic techniques. Participants are able to implement, evaluate and analyze artistic abstraction methods in computer graphics systems. They learn to implement them in Java using the Processing framework.
Contents	• 2D-Methods for the production of non-photorealistic Computer Graphics (Dithering, Halftoning, artistic screening, Stipping) • Manipulation of 2.5-dimensional Data (Images with depth), edge enhancement, depth differences, Unsharp Masking • Non-photorealistic rendering of 3D Geometry data (Computation of geometric Features, Minima and Maxima of derivations and curvatures, finding silhouettes, numerical problemes)
Course type / weekly teaching hours	lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	• Coursework: 50% of the points of the practical exercises • Examination: dependent on the number of participants oral examination (20min) or written examination (60min) • Grade: Result of the examination
Prerequisites	Requirements: experience similar to modules Mathematical Foundations for Computer Science, Informatics I and Computer Graphics and interactive Systems (or similar), elementary programming skills.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Image Processing

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students understand the core mathematical concepts behind low-level image processing, understand key algorithms for image processing and become proficient with their practical implementations.
Contents	This lecture teaches the basic principles of low-level image analysis. "Low-level" in this case does not mean it is simple or less worthy to learn, but that we are only interested in extraction of information on a more technical level, i.e. local image structure and motion without high-level scene understanding or interpretation. Furthermore, in this course we will not make use of learning-based methods, which are reserved for later lectures in the cycle. We will also study how to improve images by removing certain degradations like noise or blur, so-called inverse problems. The range of topics includes • Image filtering and the Fourier transform • Basics of pattern recognition • Edge and corner detection • Image features and sparse correspondence • Non-linear filters and image enhancement • Denoising and deconvolution • Dense image correspondence and depth reconstruction • Image sequences and motion analysis
Course type / weekly teaching hours	Lecture (3 SWS) and exercise (1 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	Written exam (or oral depending on number of participants), written exercises (weekly/bi-weekly) and active participation in exercise group.
Prerequisites	Recommended pre-requisites are basic calculus and linear algebra. Basic knowledge about probability theory and statistics will be useful. MATLAB is used in the exercises, but you can learn it as the lecture proceeds.
Language	English
Frequency of offer	At irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Immersive Analytics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The students know and understand current methods and tools in the area of Immersive Analytics. They are able to apply their knowledge in small projects.
Contents	In this course we will study, present and discuss articles concerning current methods and tools in the area of Immersive Analytics (IA). IA is an emerging research thrust investigating how new interaction and display technologies can be used to support analytical reasoning and decision making. The aim is to provide multisensory interfaces that support collaboration and allow users to immerse themselves in their data in a way that supports real-world analytics tasks. Immersive Analytics builds on technologies such as large touch surfaces, immersive virtual and augmented reality environments, sensor devices and other, rapidly evolving, natural user interface devices. Topics: - Immersive Analytics - Multisensory Immersive Analytics - Interaction for Immersive Analytics - Immersive Collaborative Analytics - Immersive Human-Centered Computational Analytics - Immersive Visual Data Stories - Situated Analytics - 3D Information Visualisation - Design Framework for Immersive Analytics - Immersive Analytics Applications Exercises support the understanding of the topic. Hands-on exercises with different technologies (3D projection systems such as MiniCave/Cave and zSpace, 3D monitorwall , HMD AR and VR systems and other technology), where you work on small projects, will complement the theoretical studies.
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS) (in form of Directed Studies)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Active participation at the discussions, exercises and the hands-on project, oral (20 min) or written (60 min) exam at the end of the course.
Prerequisites	None
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Interactive Systems

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	- Students know the basics of human information processing (e.g. perception, cognition, motor skills, mental models, mistakes). - Students know the basic rules of user interface design, can use them for established interaction styles (commands, dialogs, direct manipulation, search and browse, interactive visualizations). - Students know the basic ideas of user centered design and the fundamental methods and techniques to develop interactive systems (e.g., requirements analysis, sketching and prototyping, evaluation methods & techniques). - Students can analyze and assess existing interactive systems. - Students are able to implement basic interaction designs on their own. - Students know User Interfaces of various application areas using established interaction styles like GUIs, Web UIs, multitouch surfaces, and mobile interaction. - Students know new User Interfaces and interaction styles like Augmented Reality & Virtual Reality and how they are used in different application domains.
Contents	Interactive Systems will provide students with a comprehensive overview of the goals and research question of Human-Computer Interaction. Students gain a basic knowledge how to develop interactive systems with user requirements in mind. It covers the following topics: - Basics of human perception, cognition, and motor skills as well as mental models and mistakes - Designing usable applications that are fun to use - Basic principles of design - Established interaction styles - Basic ideas of User Centered Design - Procedure model and basic methods, techniques, and tools of usability engineering - Techniques to evaluate user interfaces Tutorials accompany the lectures and deepen the gained knowledge from a practical perspective.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	Exam: Written exam (90 minutes). Passing the tutorial is the admission requirement for the final written exam. The final grade is the grade of the written exam.
Prerequisites	None
Language	English
Frequency of offer	Summer semester

Recommended semester	1-3
Required / required elective	Required elective

Module Petri Nets

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students gain knowledge of fundamental methods for the modelling using Petri nets and for the analysis of Petri nets.
Contents	Petri nets (named after C. A. Petri) are a well-known and often used structure for the modelling and simulation of different processes in engineering, science, economy and other areas. This course presents a theory for the modelling with Petri nets and different techniques for the analysis of Petri nets. A number of examples enrich the theoretical understanding. Beside elementary Petri nets this course will also deal with Petri net extensions.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Active participation in the exercises and, depending on number of participants, oral examination (20 min) or written exam (60 min)
Prerequisites	-
Language	English
Frequency of offer	At irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Research Methods in HCI

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	At the end of this lecture, students will know - basic evaluation methods and techniques of interactive products with regards to usability and user experience - how to deduce change requirements and re-design recommendations - how to develop and define research questions and hypotheses in HCI - the crucial components of successful study designs - how to run experiments - the advantages and disadvantages of qualitative and quantitative data acquisition - how to analyze quantitative and qualitative data - pitfalls and receive tips for successful report writing
Contents	Evaluation serves the purpose to recognize usability problems early in the development phase of interactive products and develop ideas for improvement. There is a broad spectrum of techniques and methods available (e.g. observation, usability tests, surveys, etc.), which differ in many terms, such as when to apply during the development process or whether to include end-users or experts and thereby provide results for different purposes. During the theoretical part of the lecture, students will learn about the different basic methods and techniques. This includes the design and conduction of interviews, focus groups, usability tests, and inspection methods. Another goal of the lecture is to guide students in conducting experimental user studies as advanced research methods. The lecture covers the whys and hows of conducting good experiments in Human-Computer Interaction (HCI) covering both quantitative and qualitative practices. Students will learn how to build on existing work in formulating their research questions and devising hypotheses. In addition, the lecture addresses how to perform the data collection and select analysis methods that provide evidence for conclusions. Also, students learn how to narrate findings and deal with alternative explanations for results. During the practical part of the lecture, students will work on a small project in groups. They will have to conduct a study and apply the learned methods to evaluate an interactive product. This allows them to gain first hands-on experiences and also use our usability lab.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Written exam, in-class exercises, and team project
Prerequisites	No official coursework is required as a prerequisite, but students should have substantial experience with computers, especially their interactive use, and an intense interest in making them easier to use. We recommend students to be familiar with the contents of the course “Interactive Systems”.
Language	English

Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Virtual and Augmented Reality

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Successful participants of the course are able to analyze and evaluate VR and AR systems. They have advanced programming skills to create environments that construct VR and AR scenes using OpenGL, Scenegraphs and related programming environments. They have a deep understanding of technology and problems of VR systems. They are able to discuss the formation of reality in a broader perspective.
Contents	The course gives an introduction to virtual and augmented reality systems, their technology and programming. We will start with defining such systems, will go through the various systems for simulation, presentation and information, will speak about concepts for technology and what is needed to produce the illusion of reality. We will discuss augmented reality and its application in information technology. In the practical course work we will program a virtual reality system using the Oculus development system and construct VR scenes using Scenegraphs and other programing techniques. A second part of the course deals with the question of what reality is. We will discuss findings from psychology (perception), philosophy, sociology and other scientific fields to understand how reality is formed in our brain. We will see that it is absolutely not trivial to answer this question and we will discuss how modern VR and AR technology influences our perception of reality.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	oral examination (20 min) or written test (90min) will be announced at the beginning of the course
Prerequisites	basic course in Computer Graphics, OpenGl programming skills.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Description of advanced modules

In the area of advanced modules, students complete modules amounting to 36 to 66 ECTS-credits. This area comprises a seminar (3 ECTS credits), a master's project (9 ECTS credits) as well as 24 to 54 ECTS credits from modules from the curriculum of the Department of Computer and Information Science.
In the following list, advanced modules which are offered relatively regularly are included. An overview of the basic modules offered in the respective semester can be found in ZEuS.

Module Advanced Model Checking

Study programme / applicability
M. Sc. Computer and Information Science

Credits	*9*
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	It is the objective of this course to enable students to assess and use the particular model checking technologies that will be presented in this course. The participants of this course will be enabled to use model checking technologies in practical problems and to estimate and anticipate the possibilities as well as the limitations of using this technology in practical systems engineering projects. The will also be enabled to chose adequate tool support for the solution of practical model checking problems.
Contents	The course will cover advanced topics in Model Checking. The course will introduce into CTL model checking and into the foundations of BDD based symbolic model checking. The course will further cover model checking of real-time and probabilistic systems. The course Model Checking of Software and Systems is not a prerequisite for the successful completion of this course. In the excercise, different model checking tools (CTL, symbolic, timed, probabilistic) will be used to solve different kinds of model checking problems. In order to be able to write code in the different model checking languages basic programming skills are necessary.
Course type / weekly teaching hours	Lecture (4 SWS) and exercise (2 SWS)
Workload	270 hours, of which 84 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	Oral or written examination at the end of the semester Active participation in the excercises.
Prerequisites	No special model checking knowledge is needed, all foundations will be taught in the course. Basic programming skills.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Compiler Construction

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The goal of this course is to introduce the theoretical foundation as well as the practice required to construct a working compiler for a simple imperative language (Oberon-0). Students will be exposed to the different internal components of a compiler and how they work together to translate an input program into an executable. For each component, students will learn which data structures and algorithms are used to perform the corresponding stage of program translation.
Contents	A compiler is a software system that translates programs from a high-level programming language such as C, C++, or Java into the sequence of machine instructions that are executed by the actual processor. This translation is performed by the compiler in several stages. In the first two stages, the input program undergoes lexical and syntactical analysis. The resulting intermediate representation is then optimised in order to ensure efficient program execution and a compact executable. Finally, low-level code is generated. Compilers are fairly complex software systems that draw ideas and concepts from many areas of computer science. For example, program analysis is closely related to the formal notion of grammar, language, and automaton from theoretical computer science, whereas graph colouring algorithms can be used as a method for register allocation during code generation.
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	The final grade of this course will be the average of the grade of the tutorials (50%) and the grade of the written exam (50%). In order to pass the course, both of these grades need to be passing grades.
Prerequisites	The following skills, knowledge and courses are mandatory prerequisites to attend and successfully complete this course. • Computer systems: computer architecture, operating systems, etc. • System programming: students must have the ability to program in a language appropriate for system implementation. The programming assignments for this course are based on C++. • Theory of computing: languages, grammars, automatons, Chomsky hierarchy, etc. Key competences: Subversion, LaTeX, etc.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Data Mining: Advanced Topics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Based on the topics already covered in previous lectures (i.e., Data Mining: Basic Concepts), - students learn techniques to further improve the quality of training data. - students expand their knowledge in pattern mining and learn how to asses the quality of association rules. - they are able to distinguish between different neural network architectures understand their inner workings and are able to apply the algorithms to different application domains.
Contents	The students are taught advanced theoretical knowledge in data mining with accompanying practical exercises in the analysis of data. Starting from advanced data foundation problems like unbalanced training data, or the data shift problem students learn about pattern mining algorithms and measurements to judge the quality of association rules. Furthermore, different neuronal network architectures (e.g., SOMs, RNNs,…) for various data analysis tasks like clustering, or classification will be discussed.
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	A written exam at the end of the lecture.
Prerequisites	The lecture “Data Mining: Basic Concepts” is a prerequisite to attend this course.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Module Data Visualization: Advanced Topics

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The overall goal of this lecture is to teach students about the most recent approaches in information visualization. After successfully attending the lecture: - Students are able to discuss advantages and disadvantages of different visualization techniques. - Students are able to propose alternative solutions on how to display data when standard approaches do not work anymore. - Students are able to implement basic visualization techniques with D3 (data driven documents) and learn how to enrich these visualizations with additional useful features.
Contents	Advanced topics and problems in Information Visualization. The following topics are planned: - Techniques for high-dimensional data: Scatterplots, Parallel Coordinates, Projections, Pixel Visualizations) - Techniques for geo-spatial data: Flow Maps, Pixel Maps, Cartograms - Techniques for temporal data: visualization techniques for serial and periodic time-series - Techniques for relational data: node-link diagrams, treemaps - Techniques for text data - Glyphs and iconic representations - Visualization systems
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Depending on the number of participants, oral exam (of 30 minutes duration), or written exam (of 90 minutes duration). At least 50% of all points need to be gained in the exercises in order to take part in the course exam. There will be a bonus of 0.3 or 0.4 after you have passed the 80% threshold in the assignments.
Prerequisites	"Data Visualization: Basic Concepts" or "Information Visualization 1" or a comparable course on Information Visualization in another course of study are mandatory. Good programming skills in Javacript, familarity with the D3 library, and basic knowledge of databases and query languages are mandatory.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Efficient Route Planning Techniques

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	• Broad knowledge about modern route planning algorithms. • Understanding trade-offs between preprocessing time, memory consumption and preprocessing time of the different approaches. • Understanding how theoretical results can inspire better practical approaches and vice versa (algorithm engineering cycle).
Contents	In the lecture, we will discuss modern route planning algorithms for road and public transportation networks, which are used in current navigation systems, apps and web services. To accelerate the computation of shortest paths in road networks, a variety of preprocessing-based approaches (as e.g. arc-flags, hub labels or contraction hierarchies) has been develeoped in the last decade. With the help of these approaches shortest paths can be found up to a million times faster than with Dijkstra‘s algorithm. We will investigate how these approaches work in practice but also consider theoretical aspects as e.g. the complexity of the preprocessing and provable efficiency for certain graph classes. Based on these results, we will also discuss more complex route planning queries, as personlized route planning or route planning for electric vehicles.
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	Oral exam
Prerequisites	Lecture Algorithms and Data Structures
Language	English
Frequency of offer	At irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Experiments in Human-Computer Interaction

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	At the end of this course, students will know • how to develop and define research questions and hypotheses in HCI • the crucial components of successfull study designs • how to run experiments • the advantages and disadvantages of qualitative and quantitative data acquisition • how to analyse quantitative and qualitative data • pitfalls and tips for successful report writing At the end of this course, students will be able • to define a clear research goal for their thesis • chooses applicable tasks, metrics and measurements for their experiment • set up a study setting and conduct the experiment • successfully analyse and report the results
Contents	Experiments in HCI is a directed studies course designed for master students in the domain of Human-Compputer Interaction. Goal of the seminar is to guide students in conducting experimental user studies as part of their thesis. The course covers the whys and hows of conducting experiments in HCI covering both, quantitative and qualitative practices. Students will learn how to build on exisiting work in formulating their research questions and devising hypotheses. In addition, this course addresses how to perform the data collection and select analysis methods that provide evidence for conclusion. Also, students learn how to narrate findings and deal with alternative explanations for results. Based on a carefully selected reading list comprised of theoretical information about experiments, best-practice materials, and good examples, students present their own experiments and get detailed feedback on it. Additionnally, we will summarize the lessons learned to generate hand-on take-home messages.
Course type / weekly teaching hours	Directed studies (4 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study
Coursework and performance assessment	Active participation in all sessions and the presentation of a study design/setting (in-class presentation and written report) for your own research questions.
Prerequisites	This directed studies course is offered for master students writing their thesis in Interactive Systems / Human-computer interaction.
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Global Illumination Methods

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Successful participants will be able to judge and program global illumination methods in various contexts. This includes sampling strategies, texturing, anti-aliasing, geometry interaction, complex illumination scenarios.
Contents	• Raytracing methods: basic algorithm, complex geometry, acceleration strategies, material properties • Radiosity: basic method, linearization, solving the rendering equation, acceleration • further global illumination techniques in the practical course work you will implement your own raytracer
Course type / weekly teaching hours	Lecture (2 SWS) and exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	• Successful implementation of sample raytracer (groups of two are allowed) • Examination: dependent on the number of participants oral examination (20min) or written examination (60min) • Final Grade: Result of the examination
Prerequisites	Knowledge in Computer Graphics, at minimum Introduction to Computer Graphics, Programming skills (C++)
Language	English
Frequency of offer	Winter semester
Recommended semester	1-3
Required / required elective	Required elective

Module Graphical Models in Computer Vision

Study programme / applicability
M. Sc. Computer and Information Science

Credits	6
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Become familiar with the principles of graphical models and probabilistic approaches to computer vision as described above, be able to implement algorithms described in the lecture, be able to devise new solutions for computer vision problems based on the principles learned in the lecture.
Contents	Computer Vision, i.e. recovering information from images, is probabilistic in nature. Indeed, image content is usually ambiguous, requiring prior information to be able to decide for the most likely solution to any given problem. This lecture will give a systematic treatment of the probabilistic approach to computer vision, where (nearly) every problem is formulated in the Bayesian framework as finding the maximium of the posterior probability (MAP). In particular, we will investigate how to learn parameters of probability distributions, for example the distribution of pixel colors in images, and how to build more complex probability distributions with graphical (i.e. graph based) models. As such, it is a very useful first step towards “deep learning” taught in the upcoming semester, where the models will have a lot more variables.Topics include: • Introduction to probability theory and Bayesian methods • Modeling and learning probability distributions • Directed and undirected graphical models • Markov random fields • Hidden Markov models • Temporal models (Kalman filter) Applications: segmentation, depth reconstruction, tracking, …
Course type / weekly teaching hours	Lecture (3 SWS) and exercise (1 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	• Written exam (or oral depending on number of participants) • Written exercises (weekly/bi-weekly) • Active participation in exercise group
Prerequisites	Solid previous knowledge in probability/statistics useful (but basic introduction will be given), basic knowledge in Linear Algebra / Calculus recommended, basic programming skills required (Matlab will be used in exercises).
Language	English
Frequency of offer	At irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Module Interaction in Mixed Reality Spaces

Study programme / applicability
M. Sc. Computer and Information Science

Credits	*9*
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	In this course you will • gain a deeper understanding in Mixed Reality Spaces as an Interaction Paradigm • get to know application domains that can benefit from Mixed Reality Interaction • learn how to design Mixed Reality Experience • gain practical skill considering the design and implementation of mixed reality applications (including practical skills in Unity 3D development)
Contents	“Interaction in Mixed Reality Spaces” is a lecture designed for students (Bachelor, Master) in the domain of Human-Computer Interaction or Interactive Systems. In the lecture students will gain a deeper understanding in Mixed Reality (MR) technologies, their characteristics, the process of MR interface design, and possible use cases. The first part provides a theoretical introduction into the topics 3D Object Manipulation and Navigation in 3D Spaces. In the second part students will work in groups on a project using novel MR interfaces (e.g., Microsoft HoloLens, HTC Vive, iPads with ARkit, etc.).
Course type / weekly teaching hours	Lecture (4SWS)
Workload	270 hours, of which 56 hours are spent in class and 214 hours of self study.
Coursework and performance assessment	To take part on the theoretical introduction & tutorials and the practical part (development of different Mixed Reality design solutions for small projects). Exam about the theoretical part and presentation (demo) of the practical part.
Prerequisites	Basic knowledge in object-oriented programming.
Language	English
Frequency of offer	Summer semester
Recommended semester	1-3
Required / required elective	Required elective

Randomized Algorithms

Study programme / applicability
M. Sc. Computer and Information Science

Credits	*6*
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	The goal is to understand for which kind of algorithmic problems randomized approaches or analysis techniques are applicable. There are two main kinds of randomized algorithms, depending on whether the randomization impacts the solution quality or the running time, with different application fields. On examples as randomized search data structures as well as problems from discrete optimzation and computational geometry, the merits and drawbacks of these algorithms should be understood. Furthermore, the role of randomization for the design and analysis of online algorithms should be conveyed. Finally, the probabilistic method, a tool to prove the existence of certain structures using randomized construction, should be grasped such that it can be applied to new problems.
Contents	* Basic Stochastics * Las Vegas and Monte Carlo Algorithms * Randomized Search Data Structures * Getting Beyond Determinstic Lower Bounds * Sublinear Time Algorithms * Sampling * Online-Algorithms * Probabilistic Method
Course type / weekly teaching hours	Lecture (2 SWS) + exercise (2 SWS)
Workload	180 hours, of which 56 hours are spent in class and 124 hours of self study.
Coursework and performance assessment	oral exam
Prerequisites	
Language	English
Frequency of offer	At irregular intervals
Recommended semester	1-3
Required / required elective	Required elective

Master's Seminar

Study programme / applicability
M. Sc. Computer and Information Science

Credits	3
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students are able to independently prepare and hold a scientific presentation and answer questions concerning this presentation. They have a basic understanding of scientific work and the writing of scientific publications, including the correct handling of literature. They are able to write a paper on the topic, following scientific standards.
Contents	In a seminar, an academic presentation on a given topic is prepared under guidance. The presentation is held and the topic is discussed with the other participants in the seminar. Additionally, a term paper is required, which can for example have the form of a written elaboration of the presentation. For this, students receive guidance in scientific writing and practice how to use academic sources. Seminars are offered by all working groups of the faculty. The topics are taken from the following areas: - Algorithmics - Visual Computing - Database and Information Systems - Data Analysis and Visualisation - Data Mining - Theory of Computing - Life Science Informatics - Human-Computer Interaction - Multimedia Signal Processing - Software and Systems Engineering - Cyber-Physical Systems - Modelling of Complex Self-organizing Systems - Virtual Reality for Animal Collectives - Computer Vision and Image Analysis
Course type / weekly teaching hours	Seminar (2 SWS)
Workload	90 hours, of which 28 hours are spent in class and 62 hours of self study.
Coursework and performance assessment	Exact coursework is determined by the lectuerer, but coursework has to comprise at least one oral presentation and a written elaboration of the presentation..
Prerequisites	As a general rule, one or two basic or advanced courses from the respective research area. A seminar can be offered accompanying a Master's project.
Language	English or German
Frequency of offer	Each semester, a range of different seminars is offered. However, not every seminar is offered each semester
Recommended semester	2-3
Required / required elective	Required elective

Master's Project

Study programme / applicability
M. Sc. Computer and Information Science

Credits	9
Duration	one semester
Share of the module in the overall grade	varies depending on the ECTS-credits from the supplementary area of studies
Learning objectives	Students are able to carry out (minor) scientific work on their own. This may include: - finding, organizing, and systematically reading relevant research papers - analyzing, comparing, and contrasting research approaches and findings - developing research approaches - managing a comprehensive project - designing and implementing a novel method or system - conducting performance evaluations - writing an academic paper - presenting their research - answering questions and discussing their work
Contents	The project prepares the students for the writing of their Master's thesis. For this purpose the students familiarize themselves with the subject area of their thesis, e.g. through literature search and reading, evaluation of existing systems or own implementations. Projects are offered by all working groups of the faculty.
Course type / weekly teaching hours	Individual project
Workload	270 hours
Coursework and performance assessment	Is assigned by the respective lecturer
Prerequisites	Usually, a range of basic and advanced courses in the respective research area. Ideally, the corresponding seminar should be attended in advance or in parallel to the project.
Language	German or English
Frequency of offer	Each semester
Recommended semester	2-3
Required / required elective	Required elective

Final part of studies

Master’s Thesis with Colloquium

Study programme / applicability
M. Sc. Computer and Information Science

Credits	*30*
Duration	one semester
Share of the module in the overall grade	In total 40%: 30% the master’s thesis, 10% colloquium
Learning objectives	Students are able to work on a more extensive task from the field of computer and information science within the given time limit in a professional manner and according to scientific principles. They are able to present their work in a paper according to scientific standards. They can present their work in a lecture and answer questions on the content of the master's thesis and related questions on the subject area.
Contents	For the master’s thesis, students work on a topic from computer and information science on their own. Topics can be chosen from the following areas: - Algorithmics - Visual Computing - Database and Information Systems - Data Analysis and Visualisation - Data Mining - Theory of Computing - Life Science Informatics - Human-Computer Interaction - Multimedia Signal Processing - Software and Systems Engineering - Cyber-Physical Systems - Computer Vision and Image Analysis - Modelling of Complex Self-organizing Systems - Virtual Reality for Animal Collectives Depending on the chosen area and topic, students perform different tasks to work on the topic, e.g. independent literature research, evaluation of existing models, own programming work, etc..
Course type / weekly teaching hours	Self study
Workload	900 hours
Studien-/Prüfungsleistung	Writing a Master thesis and completing the colloquium
Prerequisites	- Students can only be admitted to the master’s thesis if they can document the orientation meeting and the mentor meeting and have successfully completed the performance assessments of the master’s project and seminar. - Students can only be admitted to the colloquium on the master’s thesis (thesis defence) if they have submitted the master’s thesis, have successfully completed all required coursework and performance assessments and these have been recorded in the exam administration system.
Language	German or English

Frequency of offer	Each semester
Recommended semester	4
Required / required elective	required

Supplementary courses

A maximum of 18 ECTS can be completed by supplementary courses, of which a maximum of 6 ECTS is allowed from the area of transferable skills (SQ).

Courses from other departments

Study programme / applicability
M. Sc. Computer and Information Science

Credits	0-18
Duration	Variable
Share of the module in the overall grade	Not included in the final grade
Modulnote	Not graded
Modulteile	In this module, students can complete courses from other departments of the University of Konstanz. Courses related to computer science offered by other departments can only be recognized after approval by the department's Examination Board.
Learning objectives	By attending courses from other departments, students acquire knowledge and key qualifications beyond the subject of computer science and train their thinking in interdisciplinary contexts.

Transferable skills courses

Study programme / applicability
M. Sc. Computer and Information Science

Credits	0-6
Duration	variable
Share of the module in the overall grade	Not included in the final grade
Modulnote	Not graded
Modulteile	In this module, students can attend key qualification courses offered by the Department of Computer and Information Science, by the Centre for Transferable Skills (SQ), the Language Institute (SLI) or the International Office. A subject-specific key qualification on scientific writing (Scientific Practices for Students) is obligatory if it was not completed in the previous bachelor's programme.
Learning objectives	By attending key qualification courses, students acquire knowledge and qualifications beyond the subject of computer science, which are helpful in their following course of studies and their later work life. **Learning objectives for Scientific Practices for Students:** - The students gain an overview of tools and tasks a computer scientist should be familiar with.

	- They have the ability to search and understand computer science literature and to produce effective written and oral presentations. - They know about students' responsibilities and standards in scientific writing.

Modulteil	**Scientific Practices for Students**
Contents	This course offers an introduction into scientific practices for students, especially targeted at the subject of computer science. The main goal is to aid students face the challenges of undergraduate studies and research. We will discuss a broad spectrum of questions: - What is expected of a student and how to meet these expectations? - Which tools and tasks should you be familiar with? - How to find and read computer science literature? - How to write theses? - What is plagiarism and how to avoid it? - How to design good diagrams? - How to prepare effective oral presentations?
Course type / weekly teaching hours	Seminar (2 SWS)
Workload	90 hours, of which 28 hours are spent in class and 62 hours of self study
Credits für diese Einheit	3
Coursework and performance assessment	- Active participation - homework assignments
Prerequisites	Ideally, Scientific practices should be attended in parallel to the seminar.
Language	English
Frequency of offer	Winter semester + Summer semester
Recommended semester	2-3
Required / required elective	Required (if it has not been part of the previous bachelor's degree).