JACOBS UNIVERSITY

Study Program Handbook

Computer Science

Bachelor of Science

Subject-specific Examination Regulations for Computer Science (Fachspezifische Prüfungsordnung)

The subject-specific examination regulations for Computer Science are defined by this program handbook and are valid only in combination with the General Examination Regulations for Undergraduate degree programs (General Examination Regulations = Rahmenprüfungsordnung). This handbook also contains the program-specific Study and Examination Plan (Chapter 6).

Upon graduation, students in this program will receive a Bachelor of Science (BSc) degree with a scope of 180 ECTS (for specifics see Chapter 6 of this handbook).

Version	Valid as of	Decision	Details
Fall 2021 – V1	Sep 01, 2021	March 18, 2020	V1 Originally approved by the Academic Senate
		Aug 08, 2021	V1.1 Correction of typos

Contents

1 Program Overview 5
1.1 Concept 5
1.1.1 The Jacobs University Educational Concept 5
1.1.2 Program Concept 5
1.2 Specific Advantages of Computer Science at Jacobs University 6
1.3 Program-Specific Educational Aims 7
1.3.1 Qualification Aims 7
1.3.2 Intended Learning Outcomes 8
1.4 Career Options 8
1.5 Admission Requirements 9
1.6 More Information and Contact 10
2 The Curricular Structure 10
2.1 General 10
2.2 The Jacobs University 3C Model 10
2.2.1 Year 1 – CHOICE 11
2.2.2 Year 2 – CORE 12
2.2.3 Year 3 – CAREER 13
2.3 The Jacobs Track 15
2.3.1 Methods and Skills Modules 15
2.3.2 Big Questions Modules 16
2.3.3 Community Impact Project 16
2.3.4 Language Modules 16
3 Computer Science as a Minor 18
3.1 Qualification Aims 18
3.1.1 Intended Learning Outcomes 18
3.2 Module Requirements 18
3.3 Degree 18
4 Computer Science Undergraduate Program Regulations 19
4.1 Scope of these Regulations 19
4.2 Degree 19
4.3 Graduation Requirements 19
5 Schematic Study Plan for Computer Science 20
6 Study and Examination Plan 21

7 Computer Science Module Descriptions 22
7.1 Programming in C and C++ 22
7.2 Algorithms and Data Structures 24
7.3 Introduction to Computer Science 26
7.4 Introduction to Robotics and Intelligent System 28
7.5 Databases and Web Services 30
7.6 Software Engineering 32
7.7 Operating Systems 34
7.8 Automata, Computability, and Complexity 36
7.9 Computer Networks 38
7.10 Legal and Ethical Aspects of Computer Science 40
7.11 Secure and Dependable Systems 42
7.12 Academic Skills in Computer Science 44
7.13 Computer Graphics 46
7.14 Image Processing 48
7.15 Distributed Algorithms 50
7.16 Web Application Development 52
7.17 Human-Computer Interaction 54
7.18 Artificial Intelligence 56
7.19 Robotics 58
7.20 Computer Vision 60
7.21 Machine Learning 62
7.22 Digital Design 64
7.23 Information Theory 66
7.24 Internship / Startup and Career Skills 68
7.25 Bachelor Thesis and Seminar 71
7.26 Jacobs Track Modules 74
7.26.1 Methods and Skills Modules 74
7.26.2 Big Questions Modules 85
7.26.3 Community Impact Project 110
7.26.4 Language Modules 112
8 Appendix 113
8.1 Intended Learning Outcomes Assessment-Matrix 113

1 Program Overview

1.1 Concept

1.1.1 The Jacobs University Educational Concept

Jacobs University aims to educate students for both an academic and a professional career by emphasizing four core objectives: academic quality, self-development/personal growth, internationality and the ability to succeed in the working world (employability). Hence, study programs at Jacobs University offer a comprehensive, structured approach to prepare students for graduate education as well as career success by combining disciplinary depth and interdisciplinary breadth with supplemental skills education and extra-curricular elements.

In this context, it is Jacobs University's aim to educate talented young people from all over the world, regardless of nationality, religion, and material circumstances, to become citizens of the world who are able to take responsible roles for the democratic, peaceful, and sustainable development of the societies in which they live. This is achieved through a high-quality teaching as well as manageable study loads and supportive study conditions. Study programs and related study abroad programs convey academic knowledge as well as the ability to interact positively with other individuals and groups in culturally diverse environments. The ability to succeed in the working world is a core objective for all study programs at Jacobs University, both in terms of actual disciplinary subject matter and also to the social skills and intercultural competence. Study-program-specific modules and additional specializations provide the necessary depth, interdisciplinary offerings and the minor option provide breadth while the university-wide general foundation and methods modules, mandatory German language requirements, and an extended internship period strengthen the employability of students. The concept of living and learning together on an international campus with many cultural and social activities supplements students' education. In addition, Jacobs University offers professional advising and counseling.

Jacobs University's educational concept is highly regarded both nationally and internationally. While the university has consistently achieved top marks over the last decade in Germany's most comprehensive and detailed university ranking by the Center for Higher Education (CHE), it has also been listed by the renowned Times Higher Education (THE) magazine as one of the top 300 universities worldwide (ranking group 251-300) in 2019, 2020 and 2021. The THE ranking is considered as one of the most widely observed university rankings. It is based on five major indicators: research, teaching, research impact, international orientation, and the volume of research income from industry.

1.1.2 Program Concept

Computer Science lies at the core of all modern industries and plays a major role in most areas of science as well. Computer technology changes constantly, but the fundamental principles underlying these technologies have now developed into a mature science. The Computer Science Bachelor of Science program at Jacobs University focuses on the understanding of these principles and their application in practice.

Students will obtain core computer science competencies and skills (e.g., programming and software engineering) and they will learn about fundamental abstractions and abstract notions of computing (e.g., formal languages, logic, and computability theory). They will learn about

the principles behind and the proper usage of core technologies (e.g., databases, operating systems, and computer networks). Finally, students will develop an understanding of the limitations of technology and side effects of computing systems (e.g., security, dependability, legal, and ethical aspects). Because computer science is rooted in mathematics, students will take mathematical methods modules covering calculus, linear algebra, probability theory, and numerical methods or discrete mathematics.

The job market for computer scientists has been very favorable in the last few years, and there is no indication that this will change in the near future. Because of the rapid changes in the field, it is important to focus the education on the fundamental principles, as well as, subfields of promising future relevance. Cross-disciplinary breadth and flexibility, as well as social and work organization skills are increasingly important. The minor option allows the combination of the education in computer science with a different discipline, thereby facilitating a cross-disciplinary specialization. The academic qualifications and personal profiles for academic and industrial careers differ. Jacobs University's Computer Science program responds to the needs of both areas by offering a Computer Science major designed for students who plan to work in the information technology industry or join graduate programs related to the discipline. Students choosing the minor option can acquire basic skills in a specific application domain, which makes them very well suited to work in a specific industrial sector. The minor option can also be used to obtain specific knowledge in a closely related discipline to develop a strong portfolio of knowledge at the intersection of computer science with related disciplines.

1.2 Specific Advantages of Computer Science at Jacobs University

The Computer Science program at Jacobs University aims to be rigorous with respect to the foundations, while at the same time being very contemporary with an international orientation.

- The educational approach of the faculty is to relate the theoretical contents of the discipline to their contemporary application in industry and research. The instructors aim to include recent developments of the topics covered to demonstrate how basic methods or techniques are applied today and how the material covered relates to research challenges.
- Early involvement in research projects is an essential aspect of student education. Students can obtain a vivid research experience at a very early stage, which often develops into interdisciplinary collaborations later on.
- This distinctive educational approach, together with the positive teaching environment, has been acknowledged in several rankings: In the computer science ranking published by the Centre for Higher Education (CHE) in 2015, the support by instructors and the relationship to research were ranked 1st of 68 study programs. In the European U-Multirank ranking published in 2018, the overall learning experience in computer science was ranked 10th and research-oriented teaching in computer science was ranked 2nd of 304 European universities offering Computer Science programs.
- The involvement of students and alumni in the program development process using a direct and open dialogue ensures that the program is constantly fine-tuned to the specific needs of students, such as covering certain topics at a certain time with respect to the preparation of internship or job applications.
- The program has a successful student exchange program with Carnegie Mellon University (USA). Every year, some of the best students move to Pittsburgh to study for a semester abroad at CMU. In addition, CS students have studied abroad with great success at Rice University (USA) and the University of Pennsylvania (USA).

- Student teams participate regularly in international programming competitions. Jacobs University hosted the Northwestern European Regional Contest (NWERC) of the ACM International Collegiate Programming Contest on campus in 2010 and 2011. Student teams participate in NWERC competitions since then on an annual basis. In 2014, students organized the first JacobsHack! hackathon on campus, which was sponsored, among others, by Google, Microsoft, and SAP. The 2018 edition of JacobsHack!, sponsored, among others, by Facebook, Skyscanner, GitHub and Bloomberg, attracted participants from all over Europe.

1.3 Program-Specific Educational Aims

1.3.1 Qualification Aims

The main subject-specific qualification aim is to enable students to take up qualified employment in modern industries involving information technology or to enter graduate programs related to computer science. Graduates of the Computer Science program have obtained the following competencies:

- Computer science competence

  Graduates are familiar with the theoretical foundations of computer science and they are able to design and develop computer systems addressing a given application scenario. They are able to analyze and structure complex problems and they are able to address them using methods of computer science. Graduates are able to construct and maintain complex computer systems using a structured, analytic, and creative approach.

- Communication competence

  Graduates are able to communicate subject-specific topics convincingly in both spoken and written form to fellow computer scientists or to customers.

- Teamwork and project management competence

  Graduates are able to work effectively in a team and they are able to organize workflows in complex development efforts. They are familiar with tools that support the development, testing, and maintenance of large software systems and they are able to take design decisions in a constructive way.

- Learning competence

  Graduates have acquired a solid foundation enabling them to assess their own knowledge and skills, learn effectively, and remain up to date with the latest developments in the rapidly evolving field of computer science.

- Personal and professional competence

  Graduates are able to develop a professional profile, justify professional decisions based on theoretical and methodical knowledge, and critically reflect on their behavior with respect to their consequences for society.

The design of the Computer Science program follows national guidelines published by the Gesellschaft für Informatik (GI) (GI: Empfehlungen für Bachelor- und Masterprogramme im Studienfach Informatik an Hochschulen, July 2016) and international guidelines published

jointly by the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE) (ACM/IEEE: Computer Science Curricula 2013, December 2013).

1.3.2 Intended Learning Outcomes

By the end of the program, students will be able to

- work professionally in the highly dynamic computer science field and enter graduate programs related to computer science;
- apply fundamental concepts of computer science while solving problems;
- think in an analytical way at multiple levels of abstraction;
- develop, analyze and implement algorithms using modern software engineering methods;
- understand the characteristics of a range of computing platforms and their advantages and limitations;
- choose from multiple programming paradigms, languages and algorithms to solve a given problem adequately;
- describe the fundamental theory of computation and computability;
- apply the necessary mathematical methods;
- recognize the context in which computer systems operate, including interactions with people and the physical world;
- describe the state of published knowledge in their field or a specialization within it;
- analyze and model real-life scenarios in organizations and industries using contemporary techniques of computer science, also taking methods and insights of other disciplines into account;
- appropriately communicate solutions of problems in computer science in both spoken and written form to specialists and non-specialists;
- draw scientifically founded conclusions that consider social, professional, scientific, and ethical aspects;
- work effectively in a diverse team and take responsibility in a team;
- take responsibility for their own learning, personal and professional development and role in society, reflecting on their practice and evaluating critical feedback;
- adhere to and defend ethical, scientific, and professional standards.

1.4 Career Options

Computer science is one of the key disciplines of the 21st century, which affects almost all modern industries. Consequently, the possible career paths are very broad for graduates with a computer science degree and the job market is highly favorable. The job market includes jobs such as software engineer, system integrator, information systems manager, data analyst, database administrator, application developer, cyber security analyst, IT consultant, and system analyst.

Graduates of the Computer Science program at Jacobs University have obtained positions in companies of the information technology sector such as Amazon, Cleversoft, Facebook, Google, Microsoft, SAP, Skype, 360 Treasury Systems, Twitter, Research Gate, and VMware, as well as within companies that use information technology extensively such as the BMW Group, Deutsche Bank, KPMG, and Uber. Some graduates have founded their own companies such as Deep Web Solutions GmbH, Take Off Labs, and techOS GmbH.

Past graduates have also chosen to continue their education by enrolling into graduate programs at other German universities such as the RWTH Aachen, the Technical University Berlin, and the Technical University München; at other European universities such as the University of Amsterdam, the University of Cambridge, EPFL Lausanne, the University College London, the University of Oxford, and ETH Zürich; or at international universities such as Carnegie Mellon University, Cornell University, and the University of Montreal.

The Career Services Center (CSC) as well as the Jacobs Alumni Office help students in their career development. The CSC provides students with high-quality training and coaching in CV creation, cover letter formulation, interview preparation, effective presenting, business etiquette and employer research as well as in many other aspects, thus helping students to identify and follow up rewarding careers after their time at Jacobs University. Furthermore, the Alumni Office helps students establish a long-lasting and worldwide network which is beneficial when exploring job options in academia, industry, and elsewhere.

1.5 Admission Requirements

Admission to Jacobs University is selective and based on a candidate's school and/or university achievements, recommendations, self-presentation, and performance on required standardized tests. Students admitted to Jacobs University demonstrate exceptional academic achievements, intellectual creativity, and the desire and motivation to make a difference in the world.

The following documents need to be submitted with the application:

- Recommendation Letter
- Official or certified copies of high school/university transcripts
- Educational History Form
- Standardized test results (SAT/ACT/TestAS) if applicable
- ZeeMee electronic resume (optional)
- Language proficiency test results (TOEFL, IELTS or equivalent)

German language proficiency is not required, instead all applicants need to submit proof of English proficiency.

For any student who has acquired the right to study at a university in the country where she/he has acquired the higher education entrance qualification Jacobs University accepts the common international university entrance tests as a replacement for the entrance examination. Applicants who have a subject-related entrance qualification (fachgebundene Hochschulreife) may be admitted only to respective study programs.

For more detailed information visit:

https://www.jacobs-university.de/study/undergraduate/application-information

1.6 More Information and Contact

visit our website: https://www.jacobs-university.de/study/undergraduate/programs/computer-science

2 The Curricular Structure

2.1 General

The curricular structure provides multiple elements for enhancing employability, interdisciplinarity, and internationality. The unique Jacobs Track, offered across all undergraduate study programs, provides comprehensive tailor-made modules designed to achieve and foster career competency. Additionally, a mandatory internship of at least two months after the second year of study and the possibility to study abroad for one semester give students the opportunity to gain insight into the professional world, apply their intercultural competences and reflect on their roles and ambitions for employment and in a globalized society.

All undergraduate programs at Jacobs University are based on a coherently modularized structure, which provides students with an extensive and flexible choice of study plans to meet the educational aims of their major as well as minor study interests and complete their studies within the regular period.

The framework policies and procedures regulating undergraduate study programs at Jacobs University can be found on the website (https://www.jacobs-university.de/academic-policies).

2.2 The Jacobs University 3C Model

Jacobs University offers study programs that comply with the regulations of the European Higher Education Area. All study programs are structured according to the European Credit Transfer System (ECTS), which facilitates credit transfer between academic institutions. The three-year under-graduate program involves six semesters of study with a total of 180 ECTS credit points (CP). The undergraduate curricular structure follows an innovative and student-centered modularization scheme - the 3C-Model - that groups the disciplinary content of the three study years according to overarching themes:

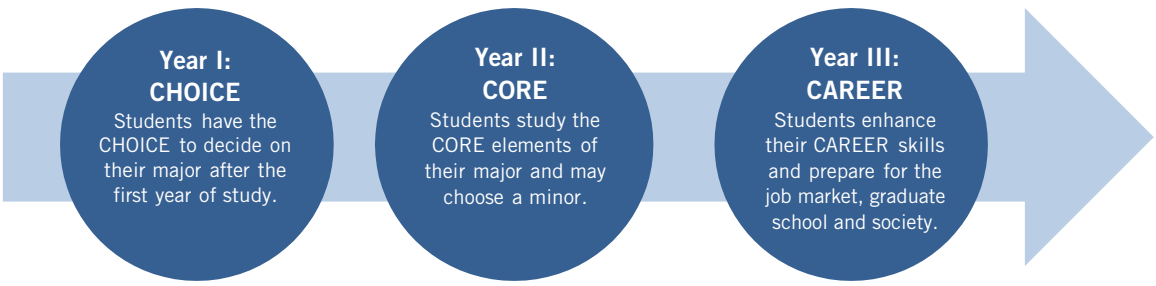

Figure 1: The Jacobs University 3C-Model

2.2.1 Year 1 – CHOICE

The first study year is characterized by a university-specific offering of disciplinary education that builds on and expands upon the students' entrance qualifications. Students select introductory modules for a total of 45 CP from the CHOICE area of a variety of study programs, of which 15-30 CP will be from their intended major. A unique feature of our curriculum structure allows students to select their major freely upon entering Jacobs University. The Academic Advising Coordinator offers curricular counseling to all Bachelor students independently of their major, while Academic Advisors support students in their decision-making regarding their major study program as contact persons from the faculty.

To pursue Computer Science as a major, the following CHOICE modules (30 CP) need to be taken as mandatory modules:

- CHOICE Module: Programming in C and C++ (7.5 CP)
- CHOICE Module: Algorithms and Data Structures (7.5 CP)
- CHOICE Module: Introduction to Computer Science (7.5 CP)
- CHOICE Module: Introduction to Robotics and Intelligent Systems (7.5 CP)

The first two modules, Programming in C and C++ and Algorithms and Data Structures, introduce students to imperative and object-oriented programming and basic algorithms and data structures. The Introduction to Computer Science module discusses abstract and concrete notions of computing machines and algorithms, and the representation of information. Students are also exposed to a pure functional programming language. The Introduction to Robotics and Intelligent Systems module addresses the interaction of computers with the physical world and it lays the foundation for intelligent systems.

The remaining CHOICE modules (15 CP) can be selected in the first year of studies according to interest and with the aim to allow a change of major up until the beginning of the second year, when the major choice becomes fixed (see 2.2.1.1 below).

2.2.1.1 Major Change Option

Students can still change to another major at the beginning of their second year of studies if they have taken the corresponding mandatory CHOICE modules in their first year of studies. All students must participate in a seminar on the major change options in the O-Week and consult their Academic Advisor in the first year of studies prior to changing their major.

Students that would like to retain a further option are strongly recommended to additionally register for the CHOICE modules of one of the following study programs in their first year. The module descriptions can be found in the respective Study Program Handbook.

- Earth and Environmental Studies (EES)
  CHOICE Module: General Earth and Environmental Sciences (7.5 CP)
  CHOICE Module: General Geology (7.5 CP)

- International Relations: Politics and History (IRPH)
  CHOICE Module: Introduction to International Relations Theory (7.5 CP)
  CHOICE Module: Introduction to Modern European History (7.5 CP)

- Integrated Social and Cognitive Psychology (ISCP)
  CHOICE Module: Essentials of Cognitive Psychology (7.5 CP)
  CHOICE Module: Essentials of Social Psychology (7.5 CP)

- Physics (Phys)
  CHOICE Module: Classical Physics (7.5 CP)
  CHOICE Module: Modern Physics (7.5 CP)

2.2.2 Year 2 – CORE

In their second year, students take a total of 45 CP from a selection of in-depth, discipline-specific CORE modules. Building on the introductory CHOICE modules and applying the methods and skills acquired so far (see 2.3.1), these modules aim to expand the students' critical understanding of the key theories, principles, and methods in their major for the current state of knowledge and best practice.

To pursue Computer Science as a major, at least the following mandatory CORE modules (30 CP) need to be taken:

- CORE Module: Databases and Web Services (7.5 CP)
- CORE Module: Software Engineering (7.5 CP)
- CORE Module: Operating Systems (7.5 CP)
- CORE Module: Automata, Computability, and Complexity (7.5 CP)

Students decide to complement their studies by taking the discipline-specific mandatory elective CORE modules (15 CP):

- CORE Module: Computer Networks (5 CP)
- CORE Module: Legal and Ethical Aspects of Computer Science (2.5 CP)
- CORE Module: Secure and Dependable Systems (5 CP)
- CORE Module: Academic Skills in Computer Science (2.5 CP)

or substitute these modules with CORE modules from other study programs according to interest and/or or with the aim of pursuing a minor in a second field.

Computer Science students can take CORE modules (or more advanced Specialization modules) from a second discipline, which allows them to incorporate a minor study track into their undergraduate education, within the 180 CP required for a bachelor's degree. The educational aims of a minor are to broaden the students' knowledge and skills, support the critical reflection of statements in complex contexts, foster an interdisciplinary approach to problem-solving, and to develop an individual academic and professional profile in line with students' strengths and interests. This extra qualification will be highlighted in the transcript.

The Academic Advising Coordinator, Academic Advisor, and the Study Program Chair of the minor study program support students in the realization of their minor selection; the consultation with the Academic Advisor is mandatory when choosing a minor.

As a rule, this requires Computer Science students to

- select two CHOICE modules (15 CP) from the desired minor program in the first year and
- substitute the mandatory elective Computer Science CORE modules Computer Networks (5 CP), Legal and Ethical Aspects of Computer Science (2.5 CP), Secure

and Dependable Systems (5), and Academic Skills in CS (2.5) in the second year with the default minor CORE modules of the minor study program. Note that the substituted CORE modules can still be selected in the third year as specialization modules.

The requirements for each specific minor are described in the handbook of the study program offering the minor (Chapter 3.2) and are marked in the respective Study and Examination Plans. For an overview of accessible minors, please check the Major/Minor Combination Matrix which is published at the beginning of each academic year.

2.2.3 Year 3 – CAREER

During their third year, students prepare and make decisions about their career path after graduation. To explore available choices and to gain professional experience, students undertake a mandatory summer internship. The third year of studies allows Computer Science students to take Specialization modules within their discipline, but also focuses on the responsibility of students beyond their discipline (see Jacobs Track).

The 5th semester also opens a mobility window for a diverse range of study abroad options. Finally, the 6th semester is dedicated to fostering the students' research experience by involving them in an extended Bachelor thesis project.

2.2.3.1 Internship / Start-up and Career Skills Module

As a core element of Jacobs University's employability approach students are required to engage in a mandatory two-month internship of 15 CP that will usually be completed during the summer between the second and third years of study. This gives students the opportunity to gain first-hand practical experience in a professional environment, apply their knowledge and understanding in a professional context, reflect on the relevance of their major to employment and society, reflect on their own role in employment and society, and find a professional orientation. The internship can also establish valuable contacts for the students' Bachelor's thesis project, for the selection of a Master program graduate school or further employment after graduation. This module is complemented by career advising and several career skills workshops throughout all six semesters that prepare students for the transition from student life to professional life. As an alternative to the full-time internship, students interested in setting up their own company can apply for a start-up option to focus on developing of their business plans.

For further information, please contact the Career Services Center (https://www.jacobs-university.de/career-services)

2.2.3.2 Specialization Modules

In the third year of their studies, students take 15 CP from major-specific or major-related, advanced Specialization modules to consolidate their knowledge and to be exposed to state-of-the-art research in the areas of their interest. This curricular component is offered as a portfolio of modules, from which students can make free selections during their 5th and 6th semester. The default specialization module size is 5 CP, with smaller 2.5 CP modules being possible as justified exceptions.

To pursue Computer Science as a major, at least 15 CP from the following mandatory elective Specialization Modules need to be taken:

- Specialization: Computer Graphics (5 CP)
- Specialization: Human Computer Interaction (5 CP)
- Specialization: Image Processing (5 CP)
- Specialization: Distributed Algorithms (5 CP)
- Specialization: Web Application Development (5 CP)
- CORE: Artificial Intelligence (5 CP)
- CORE: Robotics (5 CP)
- CORE: Machine Learning (5 CP)
- CORE: Computer Vision (5 CP)
- Specialization: Digital Design (5 CP)
- CORE: Information Theory (5 CP)

Students pursuing a minor in a second field of studies can additionally select specialization modules from the CORE modules Computer Networks (5 CP), Legal and Ethical Aspects of Computer Science (2.5 CP), Secure and Dependable Systems (5 CP), and Academic Skills in Computer Science (2.5 CP). It is highly recommended that students pursuing a minor take the modules Legal and Ethical Aspects of Computer Science (2.5 CP) and Academic Skills in Computer Science (2.5 CP).

2.2.3.3 Study Abroad

Students have the opportunity to study abroad for a semester to extend their knowledge and abilities, broaden their horizons and reflect on their values and behavior in a different context as well as on their role in a global society. For a semester abroad (usually the 5th semester), modules related to the major with a workload equivalent to 22.5 CP must be completed. Modules recognized as study abroad CP need to be pre-approved according to Jacobs University study abroad procedures. Several exchange programs allow students to directly enroll at prestigious partner institutions worldwide. Jacobs University's participation in Erasmus+, the European Union's exchange program, provides an exchange semester at a number of European universities that include Erasmus study abroad funding.

For further information, please contact the International Office (https://www.jacobs-university.de/study/international-office).

Computer Science students that wish to pursue a study abroad in their 5th semester are required to select their modules at the study abroad partners such that they can be used to substitute between 10-15 CP of major-specific Specialization modules and between 5-15 CP of modules equivalent to the non-disciplinary Big Questions modules or the Community Impact Project (see Jacobs Track). In their 6th semester, according to the study plan, returning study-abroad students complete the Bachelor Thesis/Seminar module (see next section), they take any missing Specialization modules to reach the required 15 CP in this area, and they take any missing Big Questions modules to reach 15 CP in this area. Study abroad students are allowed to substitute the 5 CP Community Impact Project (see Jacobs Track below) with 5 CP of Big Questions modules.

2.2.3.4 Bachelor Thesis/Seminar Module

This module is a mandatory graduation requirement for all undergraduate students. It consists of two module components in the major study program guided by a Jacobs faculty member: the Bachelor Thesis (12 CP) and a Seminar (3 CP). The title of the thesis will appear on the students' transcripts.

Within this module, students apply the knowledge skills, and methods they have acquired in their major discipline to become acquainted with actual research topics, ranging from the identification of suitable (short-term) research projects, preparatory literature searches, the realization of discipline-specific research, and the documentation, discussion, and interpretation of the results.

With their Bachelor Thesis students demonstrate mastery of the contents and methods of the computer science research field. Furthermore, students show the ability to analyze and solve a well-defined problem with scientific approaches, a critical reflection of the status quo in scientific literature, and the original development of their own ideas. With the permission of a Jacobs Faculty Supervisor, the Bachelor Thesis can also have an interdisciplinary nature. In the seminar, students present and discuss their theses in a course environment and reflect on their theoretical or experimental approach and conduct. They learn to present their chosen research topics concisely and comprehensively in front of an audience and to explain their methods, solutions, and results to both specialists and non-specialists.

2.3 The Jacobs Track

The Jacobs Track, an integral part of all undergraduate study programs, is another important feature of Jacobs University's educational model. The Jacobs Track runs parallel to the disciplinary CHOICE, CORE, and CAREER modules across all study years and is an integral part of all undergraduate study programs. It reflects a university-wide commitment to an in-depth training in scientific methods, fosters an interdisciplinary approach, raises awareness of global challenges and societal responsibility, enhances employability, and equips students with augmented skills desirable in the general field of study. Additionally, it integrates (German) language and culture modules.

2.3.1 Methods and Skills Modules

Methods and skills such as mathematics, statistics, programming, data handling, presentation skills, academic writing, and scientific and experimental skills are offered to all students as part of the Methods and Skills area in their curriculum. The modules that are specifically assigned to each study programs equip students with transferable academic skills. They convey and practice specific methods that are indispensable for each students' chosen study program. Students are required to take 20 CP in the Methods and Skills area. The size of all Methods and Skills modules is 5 CP.

To pursue Computer Science as major, the following Methods and Skills modules (20 CP) are mandatory

- Methods: Calculus and Elements of Linear Algebra I (5 CP)
- Methods: Calculus and Elements of Linear Algebra II (5 CP)
- Methods: Probability and Random Processes (5 CP)

For the remaining 5 CP CS students can choose between the Methods module[1]

- Methods: Numerical Methods (5 CP)

and the Mathematics CORE module:

- CORE Module: Discrete Mathematics (5 CP)

2.3.2 Big Questions Modules

The modules in the Big Questions area (10 CP) intend to broaden students' horizons with applied problem solving between and beyond their chosen disciplines. The offerings in this area comprise problem-solving oriented modules that tackle global challenges from the perspectives of different disciplinary backgrounds that allow, in particular, a reflection of acquired disciplinary knowledge in economic, societal, technological, and/or ecological contexts. Working together with students from different disciplines and cultural backgrounds, these modules cross the boundaries of traditional academic disciplines.

Students are required to take 10 CP from modules in the Area. This curricular component is offered as a portfolio of modules, from which students can make free selections during their 5th and 6th semester with the aim of being exposed to the full spectrum of economic, societal, technological, and/or ecological contexts. The size of Big Questions Modules is either 2.5 or 5 CP.

2.3.3 Community Impact Project

In their 5th semester students are required to take a 5 CP Community Impact Project (CIP) module. Students engage in on-campus or off-campus activities that challenge their social responsibility, i.e., they typically work on major-related projects that make a difference in the community life on campus, in the campus neighborhood, Bremen, or on a cross-regional level. The project is supervised by a faculty coordinator and mentors.

Study abroad students are allowed to substitute the 5-CP Community Impact Project with 5 CP of Big Questions modules.

2.3.4 Language Modules

Communication skills and foreign language abilities foster students' intercultural awareness and enhance their employability in an increasingly globalized and interconnected world. Jacobs University supports its students in acquiring and improving these skills by offering a variety of language modules at all proficiency levels. Emphasis is put on fostering the German language skills of international students as they are an important prerequisite for non-native students to learn about, explore, and eventually integrate into their host country and its professional environment. Students who meet the required German proficiency level (e.g., native speakers) are required to select modules in any other modern foreign language offered (Chinese, French or Spanish). Hence, acquiring 10 CP in language modules, with German mandatory for non-native speakers, is a requirement for all students. This curricular component is offered as a four-semester sequence of foreign language modules. The size of the Language Modules is 2.5 CP.

[1] Students who take a minor in RIS or ECE must choose Numerical Methods.

3 Computer Science as a Minor

3.1 Qualification Aims

Students obtaining a minor in Computer Science learn the basic principles of software development and modern software development processes. They acquire an understanding of how modern information systems are designed and implemented. Upon completion of the minor, they will have obtained sufficient knowledge about computer science concepts such that they can effectively work together with professionals with a Computer Science degree. Students obtaining a minor in Computer Science can help to drive digitalization processes, as they can effectively translate requirements of the field of their major into terminology and technology used by Computer Science professionals. Students majoring in a technical discipline can obtain a minor to strengthen their understanding of how to use software and hardware components effectively, thereby achieving efficient solutions for problems in their domain.

3.1.1 Intended Learning Outcomes

With a minor in Computer Science, students will be able to

- develop solutions to problems in computer science in close collaboration with computer science professionals;
- communicate requirements appropriately to their audience and understand computer science aspects of a solution;
- apply programming concepts and basic algorithms to solve software development problems of moderate complexity in an adequate way;
- understand how design choices impact the efficiency of solutions.

3.2 Module Requirements

A minor in Computer Science requires 30 CP. The default option to obtain a minor in Computer Science is marked in the Study and Examination Plan in chapter 6. It includes the following CHOICE and CORE modules:

- CHOICE Module: Programming in C and C++ (7.5 CP)
- CHOICE Module: Algorithms and Data Structures (7.5 CP)
- CORE Module: Databases and Web Services (7.5 CP)
- CORE Module: Software Engineering (7.5 CP)

Upon the consultation with the Academic Advisor and approval by the CS Study Program Coordinator, individual CORE modules from the default minor can be replaced by other advanced modules (CORE or Specialization) from the CS major.

3.3 Degree

After successful completion, the minor in Computer Science will be listed on the final transcript under PROGRAM OF STUDY and BA/BSc – [name of the major] as "(Minor: Computer Science)."

4 Computer Science Undergraduate Program Regulations

4.1 Scope of these Regulations

The regulations in this handbook are valid for all students who entered the Computer Science undergraduate program at Jacobs University in Fall 2021. In case of a conflict between the regulations in this handbook and the general Policies for Bachelor Studies, the latter applies (see http://www.jacobs-university.de/academic-policies).

In exceptional cases, certain necessary deviations from the regulations of this study handbook might occur during the course of study (e.g., change of the semester sequence, assessment type, or the teaching mode of courses).

In general, Jacobs University Bremen reserves therefore the right to change or modify the regulations of the program handbook also after its publication at any time and in its sole discretion.

4.2 Degree

Upon successful completion of the study program, students are awarded a Bachelor of Science degree in Computer Science.

4.3 Graduation Requirements

To graduate, students need to obtain 180 CP. In addition, the following graduation requirements apply:

- Students need to complete all mandatory components of the program as indicated in the Study and Examination Plan in chapter 6 of this handbook.
- Students graduating in Computer Science without a minor have to obtain
    - 20 CP in Methods and Skills modules (mathematics),
    - 90 CP in Computer Science modules, and
    - 15 CP for the Bachelor thesis and the associated seminar.
- Students graduating in Computer Science with a minor in a second discipline have to obtain
    - 20 CP in Methods and Skills modules (mathematics),
    - 75 CP in Computer Science modules, and
- Students have to obtain 15 CP for the Bachelor thesis and the associated seminar.

5 Schematic Study Plan for Computer Science

Figure 2 shows schematically the sequence and types of modules required for the study program. A more detailed description, including the assessment types, is given in the Study and Examination Plans in following section.

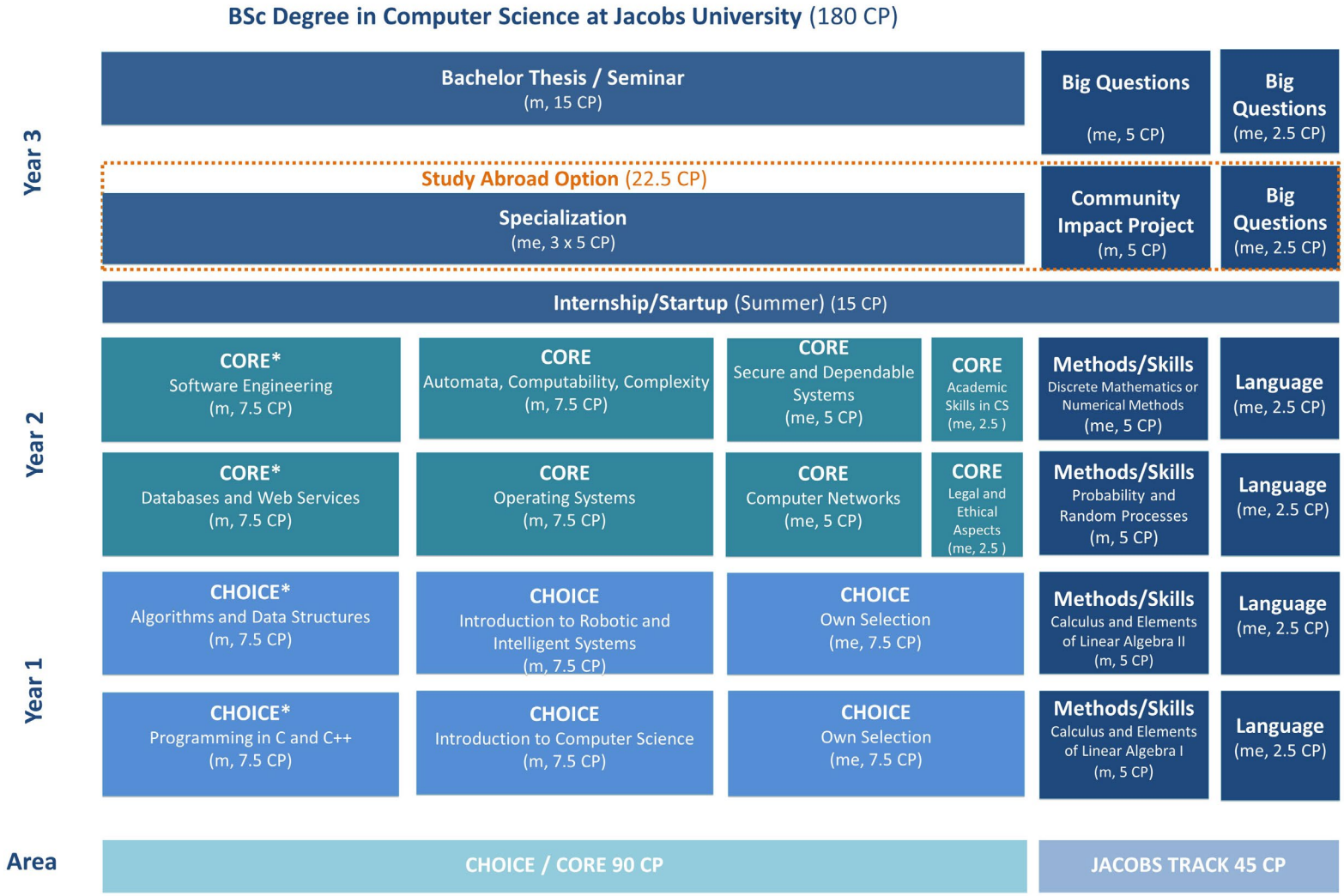

Figure 2: Schematic Study Plan

6 Study and Examination Plan

Computer Science (CS) BSc

Matriculation Fall 2021

	Program-Specific Modules	Type	Assessment	Period	Status[1]	Sem.	CP
Year 1 - CHOICE							**45**
Take the mandatory CHOICE modules listed below, this is a requirement for the Computer Science program.							
	Unit: Programming, Algorithms, and Data Structures (default minor choice modules)						**15**
CH-230	**Module: Programming in C and C++**				**m**	**1**	**7.5**
CH-230-A	Programming in C and C++	Lecture	Written examination	Examination period			2.5
CH-230-B	Programming in C and C++ Tutorial	Tutorial	Practical assessment	During the semester			5
CH-231	**Module: Algorithms and Data Structures**				**m**	**2**	**7.5**
CH-231-A	Algorithms and Data Structures	Lecture	Written examination	Examination period			
	Unit: Computer Science, Robotics, and Intelligent Systems						**15**
CH-232	**Module: Introduction to Computer Science[5]**				**m**	**1**	**7.5**
CH-232-A	Introduction to Computer Science	Lecture	Written examination	Examination period			
CH-220	**Module: Introduction to Robotics and Intelligent Systems**				**m**	**2**	**7.5**
CH-220-A	Introduction to Robotics and Intelligent Systems	Lecture	Written examination	Examination period			5
CH-220-B	Introduction to Robotics and Intelligent Systems Lab	Lab					2.5
	Unit: CHOICE (own selection)					**1/2**	**15**
Take two further CHOICE modules from those offered for all other study programs.[2]							
Year 2 - CORE							**45**
Take all CORE modules listed below or replace mandatory elective (me) modules with suitable CORE modules from other study programs.[3]							
	Unit: Advanced Computer Science I (default minor advanced modules)						**15**
CO-560	**Module: Databases and Web Services**				**m**	**3**	**7.5**
CO-560-A	Databases and Web Services - Lecture	Lecture	Written examination	Examination period			5
CO-560-B	Databases and Web Services - Project	Project	Project assessment	During the semester			2.5
CO-561	**Module: Software Engineering**				**m**	**4**	**7.5**
CO-561-A	Software Engineering	Lecture	Written examination	Examination period			2.5
CO-561-B	Software Engineering Project	Project	Project assessment	During the semester			5
	Unit: Advanced Computer Science II						**15**
CO-562	**Module: Operating Systems**				**m**	**3**	**7.5**
CO-562-A	Operating Systems	Lecture	Written examination	Examination period			
CO-563	**Module: Automata, Computability, and Complexity**				**m**	**4**	**7.5**
CO-563-A	Automata, Computability, and Complexity	Lecture	Written examination	Examination period			
	Unit: Advanced Computer Science III						**15**
CO-564	**Module: Computer Networks**				**me**	**3**	**5**
CO-564-A	Computer Networks	Lecture	Written examination	Examination period			
CO-565	**Module: Legal and Ethical Aspects of Computer Science**				**me**	**3**	**2.5**
CO-565-A	Legal and Ethical Aspects of Computer Science	Lecture	Poster presenation	Examination period			
CO-566	**Module: Secure and Dependable Systems**				**me**	**4**	**5**
CO-566-A	Secure and Dependable Systems	Lecture	Written examination	Examination period			
CO-567-A	**Module: Academic Skills in Computer Science**				**me**	**4**	**2.5**
CO-567-A	Academic Skills in Computer Science	Seminar	Project assessment	Examination period			
Year 3 - CAREER							**45**
CA-INT-900	**Module: Summer Internship**				**m**	**4/5**	**15**
CA-INT-900-0	Summer Internship		Report/Business Plan	During the 5th semester			
CA-CS-800	**Module: Thesis / Seminar CS**				**m**	**6**	**15**
CA-CS-800-T	Thesis CS	Thesis	Thesis	15th of May			12
CA-CS-800-S	Seminar CS	Seminar	Presentation	During the semester			3
	Unit: Specialization CS				**m**	**5/6**	**15**
Take a total of 15 CP Specialization Modules							
CA-S-CS-801	**Module: Computer Graphics**				**me**	**5**	**5**
CA-S-CS-801-A	Computer Graphics	Lecture	Written examination	Examination period			
CA-S-CS-802	**Module: Image Processing**				**me**	**6**	**5**
CA-S-CS-802-A	Image Processing	Lecture	Written examination	Examination period	me	6	5
CA-S-CS-803	**Module: Distributed Algorithms**				**me**	**5**	**5**
CA-S-CS-803-A	Distributed Algorithms	Lecture	Written examination	Examination period	me	5	5
CA-S-CS-804	**Module: Web Application Development**				**me**	**6**	**5**
CA-S-CS-804-A	Web Application Development	Lecture	Written examination	Examination period			2.5
CA-S-CS-804-B	Web Application Development	Project	Project assessment	During the semester			2.5
CA-S-xxx	**Specialization electives (from RIS, ECE study programs)[3]**	Lecture	Written examination	Examination period	**me**	**5/6**	**5**
Total CP							

	Jacobs Track Modules (General Education)	Type	Assessment	Period	Status[1]	Sem.	CP
Year 1 - CHOICE							**15**
	Unit: Methods / Skills						**10**
JTMS-MAT-09	**Module: Calculus and Elements of Linear Algebra I**				**m**	**1**	**5**
JTMS-09	Calculus and Elements of Linear Algebra I	Lecture	Written examination	Examination period			
JTMS-MAT-10	**Module: Calculus and Elements of Linear Algebra II**				**m**	**2**	**5**
JTMS-10	Calculus and Elements of Linear Algebra II	Lecture	Written examination	Examination period			
	Unit: Language						**5**
	German is default language. Native German speakers take modules in another offered language.						
JTLA-xxx	**Module: Language 1**				**m**	**1**	**2.5**
JTLA-xxx	Language 1	Seminar	Various	Various	me		
JTLA-xxx	**Module: Language 2**				**m**	**2**	**2.5**
JTLA-xxx	Language 2	Seminar	Various	Various	me		
Year 2 - CORE							**15**
	Unit: Methods / Skills						**10**
JTMS-MAT-12	**Module: Probability and Random Processes**				**m**	**3**	**5**
JTMS-12	Probability and Random Processes	Lecture	Written examination	Examination period			5
Take one of the two listed mandatory elective methods modules:							
CO-501	**Module: Discrete Mathematics**				**me**	**4**	**5**
CO-501-A	Discrete Mathematics	Lecture	Written examination	Examination period			
JTMS-MAT-13	**Module: Numerical Methods [4]**				**me**	**4**	**5**
JTMS-13	Numerical Methods	Lecture	Written examination	Examination period			
	Unit: Language						**5**
	German is default language. Native German speakers take modules in another offered language.						
JTLA-xxx	**Module: Language 3**				**m**	**3**	**2.5**
JTLA-xxx	Language 3	Seminar	Various	Various	me		
JTLA-xxx	**Module: Language 4**				**m**	**4**	**2.5**
JTLA-xxx	Language 4	Seminar	Various	Various	me		
Year 3 - CAREER							**15**
	Unit: Big Questions						**10**
JTBQ-BQ-xxx	**Module: Big Questions**				**m**	**5/6**	
JTBQ-xxx		Lecture	Various	Various	me		10
	Unit: Community Impact Project						**5**
JTCI-CI-950	**Module: Community Impact Project**				**m**	**5**	**5**
JTCI-950	Community Impact Project	Project	Project	Examination period			
Total CP							**180**

[1] Status (m = mandatory, me = mandatory elective)

[2] For a full listing of all CHOICE / CORE / CAREER / Jacobs Track modules please consult the CampusNet online catalogue and /or the study program handbooks.

[3] For details please see the CS program handbook

[4] Students who take a minor in Mathematics have to choose Numerical Methods module.

[5] Optional: students can also take the module in the 2nd semester

Figure 3: Study and Examination Plan

7 Computer Science Module Descriptions

7.1 Programming in C and C++

<table>
<tr><td colspan="3">Module Name
Programming in C and C++</td><td>Module Code
CH-230</td><td>Level (type)
Year 1
(CHOICE)</td><td>CP
7.5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CH-230-A | Programming in C and C++ | Lecture | 2.5
CH-230-B | Programming in C and C++ - Tutorial | Tutorial | 5</td></tr>
<tr><td colspan="2">Module Coordinator

Kinga Lipskoch</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS, RIS, and ECE</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
☒ None</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching
• Lecture attendance (17,5 hours)
• Tutorial attendance (35 hours)
• Independent study (115 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

It is recommended that students install a suitable programming environment on their notebooks. It is recommended to install a Linux system such as Ubuntu, which comes with open-source compilers such as gcc and g++ and editors such as vim or emacs. Alternatively, the open-source Code: Blocks integrated development environment can be installed to solve programming problems.</td></tr>
<tr><td colspan="6">Content and Educational Aims

This course offers an introduction to programming using the programming languages C and C++. After a short overview of the program development cycle (editing, preprocessing, compiling, linking, executing), the module presents the basics of C programming. Fundamental imperative programming concepts such as variables, loops, and function calls are introduced in a hands-on manner. Afterwards, basic data structures such as multidimensional arrays, structures, and pointers are introduced and dynamically allocated multidimensional arrays and linked lists and trees are used for solving simple practical problems. The relationships between pointers and arrays, pointers and structures, and pointers and functions are described, and they are illustrated using examples that also introduce recursive functions, file handling, and dynamic memory allocation.

The module then introduces basic concepts of object-oriented programming languages using the programming language C++ in a hands-on manner. Concepts such as classes and objects, data abstractions, and information hiding are introduced. C++ mechanisms for defining and using objects, methods, and operators are introduced and the relevance of constructors, copy constructors, and destructors for dynamically created objects is explained. Finally, concepts such as inheritance, polymorphism, virtual functions, and overloading are introduced. The learned concepts are applied by solving programming problems.</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, students will be able to

- explain basic concepts of imperative programming languages such as variables, assignments, loops, and function calls;
- write, test, and debug programs in the procedural programming language C using basic C library functions;
- demonstrate how to use pointers to create dynamically allocated data structures such as linked lists;
- explain the relationship between pointers and arrays;
- illustrate basic object-oriented programming concepts such as objects, classes, information hiding, and inheritance;
- give original examples of function and operator overloading and polymorphism;
- write, test, and debug programs in the object-oriented programming language C++.

Indicative Literature

Brian Kernighan, Dennis Ritchie: The C Programming Language, 2nd edition, Prentice Hall Professional Technical Reference, 1988.

Steve Oualline: Practical C Programming, 3rd edition, O'Reilly Media, 1997.

Bruce Eckel: Thinking in C++: Introduction to Standard C++, Prentice Hall, 2000.

Bruce Eckel, Chuck Allison: Thinking in C++: Practical Programming, Prentice Hall, 2004.

Bjarne Stroustrup: The C++ Programming Language, 4th edition, Addison Wesley, 2013.

Michael Dawson: Beginning C++ Through Game Programming, 4th edition, Delmar Learning, 2014.

Usability and Relationship to other Modules

- Mandatory for a major in CS, RIS, and ECE
- Mandatory for a minor in CS and RIS
- Pre-requisite for the CHOICE module Algorithms and Data Structures
- Elective for all other undergraduate study programs
- This module introduces the programming languages C and C++ and several other modules build on this foundation. Certain features of C++ such as templates and generic data structures and an overview of the standard template library will be covered in the Algorithms and Data Structures module.

Examination Type: Module Component Examinations

Component 1: Lecture

Assessment types: Written examination — Duration: 120 min, Weight: 33%

Scope: All theoretical intended learning outcomes of the module

Component 2: Tutorial

Assessment: Practical assessment (Programming assignments) — Weight: 67%

Scope: All practical intended learning outcomes of the module

Completion: To pass this module, the examination of each module component has to be passed with at least 45%.

7.2 Algorithms and Data Structures

<table>
<tr><td colspan="2">Module Name
Algorithms and Data Structures</td><td>Module Code
CH-231</td><td>Level (type)
Year 1
(CHOICE)</td><td>CP
7.5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CH-231-A | Algorithms and Data Structures | Lecture | 7.5</td></tr>
<tr><td>Module Coordinator

Kinga Lipskoch</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS and RIS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites

☒ Programming in C and C++</td><td>Co-requisites

☒ None</td><td>Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Class attendance (52.5 hours)
• Independent study (115 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Students should refresh their knowledge of the C and C++ programming language and be able to solve simple programming problems in C and C++. Students are expected to have a working programming environment.</td></tr>
<tr><td colspan="5">Content and Educational Aims

Algorithms and data structures are the core of computer science. An algorithm is an effective description for calculations using a finite list of instructions that can be executed by a computer. A data structure is a concept for organizing data in a computer such that data can be used efficiently. This introductory module allows students to learn about fundamental algorithms for solving problems efficiently. It introduces basic algorithmic concepts; fundamental data structures for efficiently storing, accessing, and modifying data; and techniques that can be used for the analysis of algorithms and data structures with respect to their computational and memory complexities. The presented concepts and techniques form the basis of almost all computer programs.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students will be able to
• explain asymptotic (time and memory) complexities and respective notations;
• able to prove asymptotic complexities of algorithms;
• illustrate basic data structures such as arrays, lists, queues, stacks, trees, and hash tables;
• describe algorithmic design concepts and apply them to new problems;
• explain basic algorithms (sorting, searching, graph algorithms, computational geometry) and their complexities;
• summarize and apply C++ templates and generic data structures provided by the standard C++ template library.</td></tr>
</table>

Indicative Literature

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein: Introduction to Algorithms, 3rd edition, MIT Press, 2009.

Donald E. Knuth: The Art of Computer Programming: Fundamental Algorithms, volume 1, 3rd edition, Addison Wesley Longman Publishing, 1997.

Usability and Relationship to other Modules

- Mandatory for a major in CS and RIS
- Mandatory for a minor in CS
- Pre-requisite for the following CORE modules:
    - Databases and Web Services
    - Software Engineering
    - Legal and Ethical Aspects of Computer Science
    - Computer Graphics
    - Distributed Algorithms
- Familiarity with basic algorithms and data structures is fundamental for almost all advanced modules in computer science. This module additionally introduces advanced concepts of the C++ programming language that are needed in advanced programming-oriented modules in the 2nd and 3rd years of the CS and RIS programs.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.3 Introduction to Computer Science

<table>
<tr><td colspan="2">Module Name
Introduction to Computer Science</td><td>Module Code
CH-232</td><td>Level (type)
Year 1 (CHOICE)</td><td>CP
7.5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CH-232-A | Introduction to Computer Science | Lecture | 7.5</td></tr>
<tr><td>Module Coordinator

Jürgen Schönwälder</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS, ECE and RIS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Every semester (Fall/Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class (52.5 hours)
• Independent study (115 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

It is recommended that students install a Linux system such as Ubuntu on their notebooks and that they become familiar with basic tools such as editors (vim or emacs) and the basics of a shell. The Glasgow Haskell Compiler (GHC) will be used for implementing Haskell programs.</td></tr>
<tr><td colspan="5">Content and Educational Aims

The module introduces fundamental concepts and techniques of computer science in a bottom-up manner. Based on clear mathematical foundations (which are developed as needed), the course discusses abstract and concrete notions of computing machines, information, and algorithms, focusing on the question of representation versus meaning in Computer Science.

The module introduces basic concepts of discrete mathematics with a focus on inductively defined structures, to develop a theoretical notion of computation. Students will learn the basics of the functional programming language Haskell because it treats computation as the evaluation of pure and typically inductively defined functions. The module covers a basic subset of Haskell that includes types, recursion, tuples, lists, strings, higher-order functions, and finally monads. Back on the theoretical side, the module covers the syntax and semantics of Boolean expressions and it explains how Boolean algebra relates to logic gates and digital circuits. On the technical side, the course introduces the representation of basic data types such as numbers, characters, and strings as well as the von Neuman computer architecture. On the algorithmic side, the course introduces the notion of correctness and elementary concepts of complexity theory (big O notation).</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, students will be able to

- explain basic concepts such as the correctness and complexity of algorithms (including the big O notation);
- illustrate basic concepts of discrete math (sets, relations, functions);
- recall basic proof techniques and use them to prove properties of algorithms;
- explain the representation of numbers (integers, floats), characters and strings, and date and time;
- summarize basic principles of Boolean algebra and Boolean logic;
- describe how Boolean logic relates to logic gates and digital circuits;
- outline the basic structure of a von Neumann computer;
- explain the execution of machine instructions on a von Neumann computer;
- describe the difference between assembler languages and higher-level programming languages;
- define the differences between interpretation and compilation;
- illustrate how an operating system kernel supports the execution of programs;
- determine the correctness of simple programs;
- write simple programs in a pure functional programming language.

Indicative Literature

Eric Lehmann, F. Thomson Leighton, Albert R. Meyer: Mathematics for Computer Science, online 2018.

David A. Patterson, John L Hennessy: Computer Organization and Design: The Hardware/Software Interface, 4th edition, Morgan Kaufmann, 2011.

Miran Lipovaca: Learn You a Haskell for Great Good!: A Beginner's Guide, 1st edition, No Starch Press, 2011.

Usability and Relationship to other Modules

- Mandatory for a major in CS, ECE and RIS
- Pre-requisite for the CORE modules Automata, Computability, and Complexity and Operating Systems
- This module introduces key mathematical concepts and various notions of computing machines and computing abstractions and is in particularly important for subsequent courses covering theoretical aspects of computer science. This module is also important for courses that require a basic understanding of computer architecture and program execution at the hardware level.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

Module achievement: 50% of the assignments correctly solved

This module introduces the functional programming language Haskell. Students develop their functional programming skills by solving programming problems. The module achievement ensures that a sufficient level of practical programming and problem-solving skills has been obtained.

7.4 Introduction to Robotics and Intelligent System

<table>
<tr><td colspan="3">Module Name
Introduction to Robotics and Intelligent Systems</td><td>Module Code
CH-220</td><td>Level (type)
Year 1
(CHOICE)</td><td>CP
7.5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CH-220-A | Introduction to Robotics and Intelligent Systems | Lecture | 5
CH-220-B | Introduction to Robotics and Intelligent Systems - Lab | Lab | 2.5</td></tr>
<tr><td>Module Coordinator

Francesco Maurelli</td><td colspan="3">Program Affiliation

• Robotics and Intelligent Systems (RIS)</td><td colspan="2">Mandatory Status

Mandatory for RIS, CS, ECE
Mandatory elective for Physics</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
None</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lecture (35 hours)
• Lab (17.5 hours)
• Private study (115 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Review basic linear algebra concepts, vector and matrix operations.</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module represents an initial introduction to robotics and intelligent systems, starting from the basics of mathematics and physics applied to simple robotics scenarios. It will cover transformation matrices and quaternions for reference systems. Students will then learn and the basics of trajectory planning and robotic systems. The second part of the module offers an introduction to the modeling and design of linear control systems in terms of ordinary differential equations (ODEs). Students learn how to analyze and solve systems of ODEs using state and frequency space methods. The concepts covered include time and frequency response, stability, and steady-state errors. This part culminates with a discussion on P, PI, PD, and PID controllers. The lab is designed to guide students through practical hands-on work with various components of intelligent systems. It will focus on the interfacing of a microcontroller with commonly used sensors and actuators.</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, successful students will be able to

- compute 3D transformations;
- understand and apply quaternion operations;
- apply trajectory planning techniques;
- model common mechanical and electrical systems;
- understand and apply the unilateral Laplace transform and its inverse;
- explore linear systems and tune their behavior;
- program the open-source electronic prototyping platform Arduino;
- interface Arduino to several different sensors and actuators.

Indicative Literature

R. V. Roy, Advanced Engineering Dynamics. R. V. Roy, 2015.

R. N. Jazar, Theory of Applied Robotics. Springer, 2010.

N.S. Nise, Control Systems Engineering. Wiley, 2010.

Usability and Relationship to other Modules

- Mandatory for a major in RIS, CS, ECE
- Mandatory for a minor in RIS.
- Mandatory elective for a major in Physics.
- This module is the foundation of the CORE modules in the following years.

Examination Type: Module Examination

Assessment Type: Written examination Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

Module achievement: Lab report

7.5 Databases and Web Services

<table>
<tr><td colspan="2">Module Name
Databases and Web Services</td><td>Module Code
CO-560</td><td>Level (type)
Year 2 (CORE)</td><td>CP
7.5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CO-560-A | Databases and Web Services | Lecture | 5
CO-560-B | Databases and Web Services - Project | Project | 2.5</td></tr>
<tr><td>Module Coordinator

Peter Baumann</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS
Mandatory elective for RIS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ Algorithms and Data Structures

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Project (97.5 hours)
• Independent Studies (35 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Working knowledge of basic data structures, such as trees, is required as well as familiarity with an object-oriented programming language such as C++. Basic knowledge of algebra is useful. For the project work, students benefit from having basic hands-on skills using Linux and, ideally, basic knowledge of a scripting language such as Python (the official Python documentation is available on https://docs.python.org/).</td></tr>
<tr><td colspan="5">Content and Educational Aims

This module offers a combined introduction to databases and web services. The database part starts with database design using the Entity Relationship (ER) and Unified Modeling Language (UML) models, followed by relational databases and querying them through SQL, relational design theory, indexing, query processing, transaction management, and NoSQL/Big Data databases. In the web services part, the topics addressed include markup languages, three-tier application architectures, and web services. Security aspects are addressed from both perspectives.

A hands-on group project complements the theoretical aspects: on a self-chosen topic, students implement the core of a web-accessible information system using Python (or a similar language), MySQL, and Linux, guided through homework assignments.</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, students will be able to

- read and write ER and UML diagrams;
- design and normalize data models for relational databases;
- write SQL queries and understand their evaluation by a database server;
- explain the concept of transactions and how to use transactions in application design;
- use web application frameworks to create dynamic websites;
- describe the differences of selected NoSQL data models and make a requirement-driven choice;
- restate three-tier architectures and their components;
- discuss the principles and basic mechanisms of reactive website design;
- summarize the security and privacy issues in the context of databases and web services.

Indicative Literature

Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer D. Widom: Database Systems: The Complete Book. 2nd edition, Pearson, 2008.

Ragu Ramakrishnan: Database Management Systems. 3rd edition, McGraw Hill, 2003.

James Lee: Open Source Web Development with LAMP. Pearson, 2003.

Usability and Relationship to other Modules

- Mandatory for a major in CS
- Mandatory for a minor in CS
- Serves as a mandatory elective specialization module for RIS major students.
- Pre-requisite for the CORE module Secure and Dependable Systems
- This module introduces components that are widely used by modern applications and information systems. Students can apply their knowledge in the software engineering module. This module serves as a default advanced level minor module.

Examination Type: Module Component Examinations

Module Component 1: Lecture

Assessment Type: Written examination — Duration: 120 min, Weight: 67%

Scope: All intended learning outcomes of the excluding the practical aspects

Module Component 2: Project

Assessment Type: Project — Weight: 33%

Scope: All practical aspects of the intended learning outcomes

Completion: To pass this module, the examination of each module component has to be passed with at least 45%.

7.6 Software Engineering

<table>
<tr><td colspan="3">Module Name
Software Engineering</td><td>Module Code
CO-561</td><td>Level (type)
Year 2 (CORE)</td><td>CP
7.5</td></tr>
<tr><td colspan="6">Module Component

Number | Name | Type | CP
CO-561-A | Software Engineering | Lecture | 2.5
CO-561-B | Software Engineering Project | Project | 5</td></tr>
<tr><td colspan="2">Module Coordinator

Peter Baumann</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS
Mandatory elective for RIS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Databases and Web Services

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Independent study (10 hours)
• Development work (132.5 hours)
• Exam preparation (10 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Students are expected to be able to develop software using an object-oriented programming language such as C++, and they should have access to a Linux system and associated software development tools.</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module is an introduction to software engineering and object-oriented software design. The lecture focuses on software quality and the methods to achieve and maintain it in environments of "multi-person construction of multi-version software." Based on their pre-existing knowledge of an object-oriented programming language, students are familiarized with software architectures, design patterns and frameworks, software components and middleware, Unified Modeling Language (UML)-based modelling, and validation by testing. Furthermore, the course addresses the more organizational topics of project management and version control.

The lectures are accompanied by a software project in which students have to develop a software solution to a given problem. The problem is described from the viewpoint of a customer and students working in teams have to execute a whole software project lifecycle. The teams have to create a suitable software architecture and software design, implement the components, and integrate the components. The teams have to ensure that basic quality requirements for the solution and the components are defined and satisfied. The students produce various artifacts such as design documents, source code, test cases and user documentation. All artifacts need to be maintained in a version control system and the commits should allow the instructor and other team members to track in a meaningful way the changes and who has been contributing them.</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, students will be able to

- understand and apply object-oriented design patterns;
- read and write UML diagrams;
- contrast the benefits and drawbacks of different software development models;
- design and plan a larger software project involving a team development effort;
- translate requirements formulated by a customer into computer science terminology;
- evaluate the applicability of different software engineering models for a given software development project;
- assess the quality of a software design and its implementation;
- apply tools that assist in the various stages of a software development process;
- work effectively in a team toward the goals of the team.

Indicative Literature

Ian Sommerville: Software Engineering, Pearson, 2010.

Roger Pressman: Software Engineering – a Practitioner's Approach, McGraw-Hill, 2014.

Usability and Relationship to other Modules

- Mandatory for a major in CS
- Mandatory for a minor in CS
- Serves as mandatory elective 3rd year Specialization module for RIS major students.
- Pre-requisite for the CORE module Image Processing

Examination Type: Module Component Examinations

Module Component 1: Lecture

Assessment Type: Written examination — Duration: 60 min, Weight: 33%

Scope: The first three intended learning outcomes of the module (the lecture module component)

Module Component 2: Project

Assessment Type: Project — Weight: 66%

Scope: The remaining intended learning outcomes of the module (the project module component)

Completion: To pass this module, the examination of each module component has to be passed with at least 45%.

7.7 Operating Systems

<table>
<tr><td colspan="3">Module Name
Operating Systems</td><td>Module Code
CO-562</td><td>Level (type)
Year 2 (CORE)</td><td>CP
7.5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CO-562-A | Operating Systems | Lecture | 7.5</td></tr>
<tr><td colspan="2">Module Coordinator

Jürgen Schönwälder</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ Introduction to Computer Science and Algorithms and Data Structures

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Fall)</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (52.5 hours)
• Independent study (115 hours)
• Exam preparation (20 hours)</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

187.5 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Students are expected to have a working Linux installation, which allows them to compile and run sample programs provided by the instructor and to implement their own solutions for homework assignments.</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module introduces concepts and principles used by operating systems to provide programming abstractions that enable an efficient and robust execution of application programs. Students will gain an understanding of how an operating system kernel manages hardware components and how it provides abstractions such as processes, threads, virtual memory, file systems, and inter-process communication facilities. Students learn the principles of event-driven and concurrent programming and the mechanisms that are necessary to solve synchronization and coordination problems, thereby avoiding race conditions, deadlocks, and resource starvation. The Linux kernel and runtime system will be used throughout the course to illustrate how key ideas and concepts have been implemented and how application programs can use them.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students will be able to
• explain the differences between processes, threads, application programs, libraries, and operating system kernels;
• describe well-known mutual exclusion and coordination problems;
• use semaphores to achieve mutual exclusion and solve coordination problems;
• use mutual exclusion locks and condition variables to solve synchronization and coordination problems;
• illustrate how deadlocks can be avoided, detected, and resolved;
• summarize the different mechanisms to realize virtual memory and their trade-offs;
• solve basic inter-process communication problems using signals and pipes;</td></tr>
</table>

- use socket inter-process communication primitives;
- multiplex I/O activities using suitable system calls and libraries;
- describe file system programming interfaces and the design of file systems at the operating system kernel level;
- explain how memory mapping can improve I/O performance;
- restate the functionality of a linker and the difference between static linking and dynamic linking;
- outline how different device types are supported by Unix-like kernels;
- discuss virtualization mechanisms such as containers or virtual machines.

Indicative Literature

Abraham Silberschatz, Peter B. Galvin, Greg Gagne: Applied Operating System Concepts, John Wiley, 2000.

Andrew S. Tanenbaum, Herbert Bos: Modern Operating Systems, Prentice Hall, 4th edition, Pearson, 2015.

William Stallings: Operating Systems: Internals and Design Principles, 8th edition, Pearson, 2014.

Robert Love: Linux Kernel Development, 3rd edition, Addison Wesley, 2010.

Robert Love: Linux System Programming: Talking Directly to the Kernel and C Library, 2nd edition, O'Reilly, 2013.

Usability and Relationship to other Modules

- Mandatory for a major in CS
- Pre-requisite for the CORE module Secure and Dependable Systems
- This module enables students to write programs that make efficient use of the services provided by the operating system kernel. This is particularly important for advanced modules on computer networks, robotics, and embedded systems.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module
Module achievement: 50% of the assignments correctly solved

This module includes hands-on assignments so that students can develop their system programming skills. The module achievement ensures that a sufficient level of practical system programming skills has been obtained.

7.8 Automata, Computability, and Complexity

<table>
<tr><td colspan="2">Module Name
Automata, Computability, and Complexity</td><td>Module Code
CO-563</td><td>Level (type)
Year 2 (CORE)</td><td>CP
7.5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CO-563-A | Automata, Computability, and Complexity | Lecture | 7.5</td></tr>
<tr><td>Module Coordinator

Peter Zaspel</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ Introduction to Computer Science

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (52.5 hours)
• Independent study (115 hours)
• Exam preparation (20 hours)

Workload

187.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

None</td></tr>
<tr><td colspan="5">Content and Educational Aims

This module introduces the mathematical theory of computation. Several types of abstract computational machines (called automata) are introduced together with the associated theory of formal languages. A formal language is a set of words over a defined alphabet that are well-formed according to a specific set of rules, called the grammar of the language. After studying the relationship between automata models and classes of formal languages, this course addresses the fundamental question "What problems can a computer possibly solve?" by characterizing those solvable problems, equivalently, through Turing machines, random access machines, recursive functions and lambda calculus. A full answer to the related question, "How many computational resources are needed for solving a given problem?" is not known today. However, the basic outlines of today's theory of computational complexity will be presented up to the most famous open problem in computer science, namely the "P = NP" question: if a computer could guess the right answer to a computational problem (and only needs to check its correctness), would that computer be faster than another one that cannot guess the right solution? This may seem to be a ridiculously obvious case of a clear YES answer, but in fact it is considered by many to be the deepest open question in contemporary mathematics (and computer science, of course).

This module provides the core education in theoretical computer science. The material covered in this module gives students access to any field in computer science, which is based on discrete-mathematical formal foundations, such as the theory of automata and formal languages or compiler design.</td></tr>
</table>

Intended Learning Outcomes

By the end of this module, students will be able to

- explain discrete automata models (finite state machines, pushdown automata, Turing machines);
- describe the Chomsky hierarchy of formal languages and classify formal languages;
- characterize classes of formal languages by automata models and grammars;
- define formal models of computation such as Turing machines;
- explain the equivalences of formal models of computation;
- illustrate the nature and impact of the Church–Turing hypothesis;
- construct diagonalization arguments;
- give examples of functions that are not computable;
- contrast central complexity classes (L, P, NP, EXP, …);
- apply reduction techniques both for decidability and complexity;
- create a reduction-based check of whether a problem is NP-complete.

Indicative Literature

Michael Sipser: Introduction to the Theory of Computation, 2nd edition, PWS Publishing Company, 1997. (Primary Literature).

John Hopcroft, Rajeev Motwani, Jeffrey Ullman: Introduction to Automata Theory, Languages, And Computation, 3rd edition, Pearson, 2006.

Usability and Relationship to other Modules

- Mandatory for a major in CS
- This module provides the core education in theoretical computer science.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.9 Computer Networks

<table>
<tr><td colspan="2">Module Name
Computer Networks</td><td>Module Code
CO-564</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CO-564-A | Computer Networks | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Jürgen Schönwälder</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ Algorithms and Data Structures

Co-requisites
☒ Operating Systems

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Students are expected to be familiar with the C programming language and to learn basics of higher-level scripting languages such as Python (the official Python documentation is available on https://docs.python.org/).</td></tr>
<tr><td colspan="5">Content and Educational Aims

Computer networks such as the Internet play a critical role in today's connected world. This module discusses the technology of Internet services in depth to enable students to understand the core issues involved in the design of modern computer networks. Fundamental algorithms and principles are explained in the context of existing protocols as they are used in today's Internet. Students taking this course should finally understand the technical complexity behind everyday online services such as Google or YouTube.

Students taking this module will understand how computer networks work and they will be able to assess communication networks, including aspects such as performance but also robustness and security. Students will learn that the design of communication networks is not only influenced by technical constraints but also by the necessity to define common standards, which often requires to take engineering decisions that reflect non-technical requirements.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students will be able to
• recall layering principles and the OSI reference model;
• articulate the organization of the Internet and the organization involved in providing Internet services;
• describe media access control, flow control, and congestion control mechanisms;
• explain how local area networks differ from global networks;
• illustrate how frames are forwarded in local area networks;
• contrast addressing mechanisms and translations between addresses used at different layers;</td></tr>
</table>

- demonstrate how the Internet network layer forwards packets;
- present how routing algorithms and protocols are used to determine and select routes;
- describe how the Internet transport layer provides different end-to-end services;
- demonstrate how names are resolved to addresses and vice versa;
- summarize how application layer protocols send and access electronic mail or access resources on the world-wide web;
- design and implement simple application layer protocols;
- recognize to which extent computer networks are fragile and evaluate strategies to cope with the fragility;
- analyze traffic traces produced by a given computer network.

Indicative Literature

James F. Kurose, Keith W. Ross: Computer Networking: A Top-Down Approach Featuring the Internet, 3rd Edition, Addison-Wesley, 2004.

Andrew S. Tanenbaum: Computer Networks, 4th Edition, Prentice Hall, 2002.

Usability and Relationship to other Modules

- Mandatory elective module for a major in CS
- Pre-requisite for the CORE module Secure and Dependable Systems
- The module should be taken together with the module Operating Systems, because a significant portion of the communication technology is implemented at the operating system level. An understanding of operating system concepts and abstractions will help students to understand how computer network technology is commonly implemented and made available to applications. The specialization module Distributed Algorithms discusses algorithms for solving problems commonly found in distributed systems that use computer networks to exchange information. The module Secure and Dependable Systems introduces cryptographic mechanisms that can be used to secure communication over computer networks.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min, Weight: 100%

Scope: All intended learning outcomes of the module

7.10 Legal and Ethical Aspects of Computer Science

Module Name	*Module Code*	*Level (type)*	*CP*
Legal and Ethical Aspects of Computer Science	CO-565	Year 2 (CORE)	2.5

Module Components

Number	*Name*	*Type*	*CP*
CO-565-A	Legal and Ethical Aspects of Computer Science	Lecture	2.5

Module Coordinator	*Program Affiliation*	*Mandatory Status*
N. N.	• Computer Science (CS)	Mandatory for CS

Entry Requirements			*Frequency*	*Forms of Learning and Teaching*
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	• Class attendance (17.5 hours) • Private study (35 hours) • Poster preparation (10 hours)
☒ None	☒ None			
			Duration	*Workload*
			1 semester	62.5 hours

Recommendations for Preparation

None

Content and Educational Aims

Information technology has a profound impact on society. This module introduces the legal and ethical frameworks that are relevant for computer scientists taking up qualified employment or joining advanced study programs leading to a career in education and research. The module provides an overview of intellectual property rights and their regulations, data protection regulations, and ethical frameworks defined by professional organizations. Students are confronted with a collection of case studies to develop sensitivity to legal and ethical dilemmas with which people are sometimes faced during the construction or operation of advanced information processing systems.

Intended Learning Outcomes

By the end of this module, students will be able to

- recall principles of data protection regulations such as the European General Data Protection Regulation (GDPR);
- identify components of an IT system managing sensitive data that needs protection;
- summarize regulations concerning intellectual property rights;
- analyze the applicability of different closed-source and open-source software licensing models;
- describe computer science ethics and ethical frameworks defined by professional organizations;
- illustrate ethical dilemma resulting from the use of information processing systems;
- discuss the interplay of legal frameworks and ethical principles and the design of information processing systems.

Indicative Literature

Not specified.

Usability and Relationship to other Modules

- Mandatory elective module for a major in CS.

Examination Type: Module Examination

Assessment Type: Poster presentation — Duration: 10 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.11 Secure and Dependable Systems

<table>
<tr><td colspan="3">Module Name
Secure and Dependable Systems</td><td>Module Code
CO-566</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CO-566-A | Secure and Dependable Systems | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Jürgen Schönwälder</td><td colspan="3">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory for CS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Operating Systems

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Independent study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

None</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module introduces students to the fundamentals of computer security and techniques used to build and analyze dependable systems. This is an important topic given that computer systems are increasingly embedded in everyday objects (such as light bulbs) and taking over important control functions (such as driving cars). Furthermore, computer systems control complex communication systems that form critical infrastructure of the modern globalized world. Proper protection of information requires an applied understanding of cryptography and how cryptographic primitives are used to secure data and information exchanges. The aim of this module is to make students aware of what types of security vulnerabilities may arise in computing systems and how to prevent, identify, and fix them.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students will be able to

• recall dependability terminology and concepts;
• explain control flow attacks and injection attacks and defense mechanisms;
• describe network data plane and control plane attacks and defense mechanisms;
• understand symmetric and asymmetric cryptographic algorithms;
• explain how digital signatures and public key infrastructures work;
• analyze key exchange protocols for weaknesses;
• describe secure network protocols (e.g., PGP, TLS, and SSH);
• recall anonymity terminology and concepts;
• discuss information hiding mechanisms (e.g., steganography, and watermarking);
• illustrate anonymization techniques (mixes, onion routing);</td></tr>
</table>

Indicative Literature

Bruce Schneier: Applied Cryptography, 20th Anniversary Edition, Wiley, 2015.

Wm.A. Conklin, Gregory White: Principles of Computer Security, 5th Edition, McGraw-Hill, 2018.

Simon Singh: The Code Book: Science of Secrecy from Ancient Egypt to Quantum Cryptography, Anchor Books, 2000.

Usability and Relationship to other Modules

- Mandatory elective module for a major in CS.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.12 Academic Skills in Computer Science

Module Name	*Module Code*	*Level (type)*	*CP*
Academic Skills in Computer Science	CO-567	Year 2 (CORE)	2.5

Module Components

Number	*Name*	*Type*	*CP*
CO-567-A	Academic Skills in Computer Science	Seminar	2.5

Module Coordinator	*Program Affiliation*	*Mandatory Status*
Sergey Kosov	• Computer Science (CS)	Mandatory elective for CS

Entry Requirements

Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*
☒ None	☒ None	☒ None

Frequency

Annually (Spring)

Forms of Learning and Teaching

- Class attendance (17.5 hours)
- Private study (25 hours)
- Presentation / poster preparation (20 hours)

Duration

1 semester

Workload

62.5 hours

Recommendations for Preparation

None

Content and Educational Aims

This module introduces students to basic skills in reading, understanding, and evaluating scientific articles, and in presenting scientific results in presentations and publications. During the seminar, students will study some classic computer science papers with a special focus on how the papers are organized, written and how they present scientific results. Students will develop and discuss guidelines for effective writing and they will learn about techniques and tools that can be used to effectively search for literature relevant to a certain topic. Finally, students will be introduced to peer review processes.

As a project, students will emulate the workflow of a scientific conference to demonstrate the academic skills they have learned.

Intended Learning Outcomes

By the end of this module, students will be able to

- effectively find research literature for a given topic;
- critically read and assess research papers;
- present a research result in the structure of a scientific paper;
- describe how scientific peer review processes work;
- orally communicate research results effectively to a scientific community;
- describe common pitfalls in the presentation of data, algorithms, or math;
- discuss ethical issues and guidelines related to scientific publications.

Indicative Literature

Peter Zobel: Writing for Computer Science, 3rd edition, Springer, 2014.

Usability and Relationship to other Modules

- Mandatory elective for a major in CS.

Examination Type: Module Examination

Assessment Type: Project	Weight: 100%
Scope: All intended learning outcomes of the module	

7.13 Computer Graphics

Module Name	*Module Code*	*Level (type)*	*CP*
Computer Graphics	CA-S-CS-801	Year 3 (CAREER - Specialization)	5

Module Components

Number	*Name*	*Type*	*CP*
CA-S-CS-801	Computer Graphics	Lecture	5

Module Coordinator	*Program Affiliation*	*Mandatory Status*
Sergey Kosov	• Computer Science (CS)	Mandatory elective for CS and RIS

Entry Requirements			*Frequency*	*Forms of Learning and Teaching*
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	• Class attendance (35 hours) • Private study (70 hours) • Exam preparation (20 hours)
☒ Algorithms and Data Structures	☒ None			
			Duration	*Workload*
			1 semester	125 hours

Recommendations for Preparation

None

Content and Educational Aims

This module deals with the digital synthesis and manipulation of visual content. The creation process of computer graphics spans from the creation of a three-dimensional (3D) scene to displaying or storing it digitally. Prominent tasks in computer graphics are geometry processing, rendering, and animation. Geometry processing is concerned with object representations such as surfaces and their modeling. Rendering is concerned with transforming a model of the virtual world into a set of pixels by applying models of light propagation and sampling algorithms. Animation is concerned with descriptions of objects that move or deform over time. This is an introductory module covering the concepts and techniques of 3D (interactive) computer graphics. It covers mathematical foundations, basic algorithms and principles, and some advanced methods and concepts. An introduction to the implementation of simple programs using a mainstream computer graphics library completes this module.

Intended Learning Outcomes

By the end of this module, students will be able to

- construct 3D geometry representations;
- apply 3D transformations;
- understand the algorithms and optimizations applied by graphics rendering systems;
- explain the stages of modern computer graphics programmable pipelines
- implement simple computer graphics applications using graphics frameworks such as OpenGL;
- illustrate the techniques used to create animations.

Indicative Literature

John Hughes, Andries van Dam, Morgan McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akeley, Computer Graphics - Principles and Practice, 3rd edition, Addison-Wesley, 2013.

Peter Shirley, Steve Marschner, Fundamentals of Computer Graphics, 4th edition, Taylor and Francis Ltd, 2016.

Matt Pharr, Wenzel Jakob, Greg Humphreys, Physically Based Rendering: From Theory to Implementation, 3rd edition, Morgan Kaufmann, 2016.

Usability and Relationship to other Modules

- Mandatory elective for a major in CS.
- Serves as a 3rd year specialization module for RIS major students.
- Students with a strong interest in graphical user interfaces are encouraged to also select the Human–Computer Interaction specialization module, which discusses among other things how computer graphics can be used as a component of interactive graphical user interfaces.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.14 Image Processing

<table>
<tr><td colspan="2">Module Name
Image Processing</td><td>Module Code
CA-S-CS-802</td><td>Level (type)
Year 3 (CAREER - Specialization)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CA-CS-802 | Image Processing | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Horst Hahn</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory elective for CS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Algorithms and Data Structures

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• None</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

None</td></tr>
<tr><td colspan="5">Content and Educational Aims

The module provides a foundation of the theory and applications of digital image processing. The first part concentrates on morphological image processing, which is one of the most basic yet powerful tool sets in dealing with digital images, and it is the backbone of many of today's high-performance image analysis systems. The module starts by introducing concepts such as dilation, erosion, geodesic transformations, morphological filtering, and the watershed transform. It then develops into advanced strategies for image segmentation and texture analysis. The second part of the module will concentrates on understanding problems from real-world applications, such as in biomedical imaging, and provides an overview of the broader field of image processing. The course can be combined with other courses on machine learning and signal analysis. Homework assignments will cover C/C++ implementations of basic and combined image processing algorithms.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students will be able to
• explain the theory and concepts of image processing;
• illustrate concepts such as dilation, erosion, geodesic transformations, and morphological filtering;
• analyze image segmentation and texture analysis algorithms;
• design and implement their own image processing algorithms in C/C++.</td></tr>
<tr><td colspan="5">Indicative Literature

Milan Sonka, Vaclav Hlavac, Roger Boyle: Image Processing, Analysis, and Machine Vision, 3rd edition, Nelson Engineering, 2007.</td></tr>
</table>

Pierre Soille, Morphological Image Analysis: Principles and Applications, 2nd edition, Springer, 2004.
Usability and Relationship to other Modules • Mandatory elective for a major in CS
Examination Type: Module Examination Assessment Type: Written examination — Duration: 120 min Weight: 100% Scope: All intended learning outcomes of the module

7.15 Distributed Algorithms

<table>
<tr><td colspan="2">Module Name
Distributed Algorithms</td><td>Module Code
CA-S-CS-803</td><td>Level (type)
Year 3 (Specialization)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CA-S-CS-803 | Distributed Algorithms | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Kinga Lipskoch</td><td colspan="2">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory elective for CS and RIS.</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites: ☒ Algorithms and Data Structures
Co-requisites: ☒ None
Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Fall or Spring)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

None</td></tr>
<tr><td colspan="5">Content and Educational Aims

Distributed algorithms are the foundation of modern distributed computing systems. They are characterized by a lack of knowledge of a global state, a lack of knowledge of a global time, and inherent non-determinism in their execution. The course introduces basic distributed algorithms using an abstract formal model, which is centered on the notion of a transition system. The topics covered are logical clocks, distributed snapshots, mutual exclusion algorithms, wave algorithms, election algorithms, reliable broadcast algorithms, and distributed consensus algorithms. Process algebras are introduced as another formalism to describe distributed and concurrent systems.

The distributed algorithms introduced in this module form the foundation of computing systems that have to be scalable and fault-tolerant, e.g., large-scale distributed non-standard databases or distributed file systems. The course is recommended for students interested in the design of scalable distributed computing systems.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students will be able to

• describe and analyze distributed algorithms using formal methods such as transition systems;
• explain different algorithms to solve election problems;
• illustrate the limitations of time to order events and how logical clocks and vector clocks overcome these limitations;
• apply distributed algorithms to produce consistent snapshots of distributed computations;
• describe the differences among wave algorithms for different topologies;
• analyze and implement distributed consensus algorithms such as Paxos and Raft;
• use a process algebra such as communicating sequential processes or -calculus to model distributed algorithms.</td></tr>
<tr><td colspan="5">Indicative Literature</td></tr>
</table>

Maarten van Steen, Andrew S. Tanenbaum: Distributed Systems, 3rd edition, Pearson Education, 2017.

Nancy A. Lynch: Distributed Algorithms, Morgan Kaufmann, 1996.

Usability and Relationship to other Modules

- Mandatory elective 3rd Specialization module for CS and RIS major students.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.16 Web Application Development

<table>
<tr><td colspan="3">Module Name
Web Application Development</td><td>Module Code
CA-S-CS-804</td><td>Level (type)
Year 3
(CAREER - Specialization)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CA-CS-804-A | Web Application Development | Lecture | 2.5
CA-CS-804-B | Web Application Development - Project | Project | 2.5</td></tr>
<tr><td>Module Coordinator

N.N.</td><td colspan="3">Program Affiliation

• Computer Science (CS)</td><td colspan="2">Mandatory Status

Mandatory elective for CS and Mandatory elective for RIS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Databases and Web Services

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (17.5 hours)
• Private study (40 hours)
• Project work (50 hours)
• Exam preparation (17.5 hours)

Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

None</td></tr>
<tr><td colspan="6">Content and Educational Aims

A web application is a client-server computer program where the client provides the user interface and the client side logic runs in a web browser or as an app running on a mobile device such as a smart phone or a tablet. A key characteristic is that more complex application logic and data storage is realized by a server offering a web application programming interface.

This module focuses on the client side of web application and introduces technologies that can be used to implement interactive user interfaces and client side logic. It builds on the module databases and web services, which covers the data storage components and server side logic of web applications.

This module consists of a lecture and an associated project. The lecture component introduces programming languages and frameworks that are widely used for implementing the client side of web applications such as Java, Kotlin, Swift, JavaScript and frameworks built on top of them. In the project component, students develop web applications and test them on existing and openly accessible web services.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students will be able to

• explain the document object model behind HTML and its relation to CSS;
• discuss the principles and basic mechanisms of reactive website design;</td></tr>
</table>

- analyze the interactions between web applications and web services.
- use languages such as Java, Kotlin, or Swift to implement mobile web applications;
- use web standards such as HTML, CSS, and JavaScript to implement web applications running in standard web browsers.

Indicative Literature

Stoyan Stefanov: JavaScript Patterns, O'Reilly Media, 2010.

Alexey Soshin: Hands-on Design Patterns with Kotlin, Packt Publishing, 2018.

Alex Banks, Eve Porcello: Learning React: Functional Web Development.with React and Flux, O'Reilly, 2017.

Usability and Relationship to other Modules

- Mandatory elective for a major in CS.
- Mandatory elective for a major in RIS.

Examination Type: Module Component Examinations

Module Component 1: Lecture

Assessment Type: Written examination — Duration: 120 min, Weight: 50%

Scope: First group of intended learning outcomes of the module

Module Component 2: Project

Assessment Type: Project — Weight: 50%

Scope: Second group of intended learning outcomes of the module

Completion: To pass this module, the examination of each module component has to be passed with at least 45%.

7.17 Human-Computer Interaction

Module Name	*Module Code*	*Level (type)*	*CP*
Human Computer Interaction	CA-S-RIS-802	Year 3 (CAREER-Specialization)	5

Module Components

Number	*Name*	*Type*	*CP*
CA-RIS-802	Human Computer Interaction	Lecture	5

Module Coordinator	*Program Affiliation*	*Mandatory Status*
Sergey Kosov	• Robotics and Intelligent Systems (RIS)	Mandatory elective for RIS and CS

Entry Requirements			*Frequency*	*Forms of Learning and Teaching*
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	• Class attendance (35 hours) • Private study (70 hours) • Exam preparation (20 hours)
☒ None	☒ None	• None	*Duration*	*Workload*
			1 semester	125 hours

Recommendations for Preparation

None

Content and Educational Aims

Computer systems often interact with human beings. The design of a good human–computer interface is often crucial for the acceptance and the success of a software system. Human–computer interface designs have to satisfy several requirements such as usability, learnability, efficiency, accessibility, and safety. The module discusses the evolution of human–computer interaction models and introduces design principles for graphical user interfaces and other types of interaction (e.g., visual, voice, gesture). Human–computer interaction designs are often evaluated using prototypes or mockups that can be given to test candidates to evaluate the effectiveness of the design. The module introduces evaluation strategies as well as tools and techniques that can be used to prototype human–computer interfaces.

Intended Learning Outcomes

By the end of this module, students should be able to

- explain the evolution of human–computer interaction models;
- design and implement simple graphical user interfaces;
- explain ergonomic principles guiding the design of user interfaces;
- illustrate different types of interaction (e.g., visual, voice, gestures) and their usability aspects;
- evaluate aspects of and tradeoffs between usability, learnability, efficiency, and safety;
- apply scientific methods to evaluate interfaces with respect to their usability and other desirable properties;
- use prototyping tools that can be employed to create mockups of user interfaces during the early stages of a software project.

Indicative Literature

Not specified

Usability and Relationship to other Modules

- Students with a strong interest in graphical user interfaces are encouraged to also select the Computer Graphics specialization module, which introduces methods and technologies for creating computer graphics and animations.
- Mandatory elective third year Specialization module for CS and RIS major students.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min, Weight: 100%

Scope: All intended learning outcomes of the module

7.18 Artificial Intelligence

<table>
<tr><td colspan="3">Module Name
Artificial Intelligence</td><td>Module Code
CO-547</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CO-547-A | Artificial Intelligence | Lecture | 5</td></tr>
<tr><td colspan="2">Module Coordinator

Andreas Birk</td><td colspan="2">Program Affiliation

• Robotics and Intelligent Systems (RIS)</td><td colspan="2">Mandatory Status

Mandatory for RIS
Mandatory elective for CS</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ Programming in C/C++
☒ Introduction to RIS

Co-requisites
☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Spring)</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Revise content of the pre-requisite modules.</td></tr>
<tr><td colspan="6">Content and Educational Aims

Artificial Intelligence (AI) is an important subdiscipline of Computer Science that deals with technologies to automate the performance of tasks that are usually associated with intelligence. AI methods have a significant application potential, as there is an increasing interest and need to generate artificial systems that can carry out complex missions in unstructured environments without permanent human supervision. The module teaches a selection of the most important methods in AI. In addition to general-purpose techniques and algorithms, it also includes aspects of methods that are especially targeted for physical systems such as intelligent mobile robots or autonomous cars.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students should be able to
• outline and explain the history, general developments, and application areas of AI;
• apply the basic concepts and methods of behavior-oriented AI;
• use concepts and methods of search algorithms for problem-solving;
• explain the basic concepts of path-planning as an application example for domain-specific search;
• apply basic path-planning algorithms and to compare their relations to general search algorithms;
• write and explain concepts of propositional and first-order logic;
• use logic representations and inference for basic examples of artificial planning systems.</td></tr>
<tr><td colspan="6">Indicative Literature

S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, 2009.

S. M. LaValle, Planning Algorithms. Cambridge University Press, 2006.

J.-C. Latombe, Robot Motion Planning, Springer, 1991.</td></tr>
</table>

Usability and Relationship to other Modules

- This module gives an introduction to Artificial Intelligence (AI) excluding the aspects of machine learning (ML), which are covered in a dedicated module that complements this one.
- Mandatory for a major in RIS
- This module serves as a third year Specialization module for CS major students.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.19 Robotics

<table>
<tr><td colspan="2">Module Name
Robotics</td><td>Module Code
CO-540</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CO-540-A | Robotics | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Andreas Birk</td><td colspan="2">Program Affiliation

• Robotics and Intelligent Systems (RIS)</td><td colspan="2">Mandatory Status

Mandatory for RIS
Mandatory elective for CS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites

☒ Programming in C/C++
☒ Introduction to RIS

Co-requisites

☒ None

Knowledge, Abilities, or Skills</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Revise content of the pre-requisite modules.</td></tr>
<tr><td colspan="5">Content and Educational Aims

Robotics is an area that is driven by dreams from science fiction and the reality of engineering. The module intends to provide an understanding of the formal foundations of this area as well as its technological state of the art and future directions. The course accordingly gives an introduction to the core algorithmic, mathematical, and engineering concepts and methods of robotics. This includes concepts and methods that are used for well-established tools of factory automation, especially in the form of robot-arms, as well as increasingly relevant intelligent mobile systems such as autonomous cars or autonomous transport systems.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students should be able to
• outline and explain the history, general developments, and application areas of robotics;
• apply the concepts and methods to describe space and motions therein including homogeneous coordinates and transforms as well as quaternions;
• use the spatial concepts and methods for the forward kinematics (FK) of robot-arms;
• explain basic concepts of simple actuators, including electrical motors and gear systems;
• apply concepts and methods to derive the inverse kinematics of robot-arms and related systems such as legs in analytical and numerical forms;
• apply concepts and methods of wheeled locomotion including FK and IK of the differential and of the omni-directional drive;
• use basic concepts and methods of dynamics;
• Explain and use core concepts and methods of global localization, e.g., multilateration and multidimensional scaling;
• use the basic concepts and methods of error propagation estimation in the context of relative localization with dead-reckoning;</td></tr>
</table>

- outline and compare the basic concepts and methods of mapping.

Indicative Literature

J. J. Craig, Introduction to robotics - Mechanics and control, Prentice Hall, 2005.
G. Dudek and M. Jenkin, Computational Principles of Mobile Robotics, Cambridge University Press, 2000.
R. Siegwart and I. R. Nourbakhsh, Introduction to Autonomous Mobile Robots, The MIT Press, 2004.
S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics, MIT Press, 2005.
H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, Principles of Robot Motion, MIT Press, 2005.

Usability and Relationship to other Modules

- Mandatory for a major in RIS
- Mandatory for a minor in RIS
- This module serves as a third Year Specialization module for CS major students.
- This module gives an introduction to Robotics, which is a core discipline of Robotics and Intelligent System (RIS) and an important area of possible future employment.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.20 Computer Vision

<table>
<tr><td colspan="2">Module Name
Computer Vision</td><td>Module Code
CO-546</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CO-546-A | Computer Vision | Lecture/lab | 5</td></tr>
<tr><td>Module Coordinator

Francesco Maurelli</td><td colspan="2">Program Affiliation

• Robotics and Intelligent Systems (RIS)</td><td colspan="2">Mandatory Status

Mandatory elective for RIS
Mandatory elective for CS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ Intro to RIS
☒ Programming in C/C++

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• Basic knowledge of robotics middleware (RIS Lab I)</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Refresh basic programming skills in MATLAB and/or Python</td></tr>
<tr><td colspan="5">Content and Educational Aims

Computer Vision algorithms are used in a variety of real-world applications that include surveillance and object tracking, 3D model building (photogrammetry), and object recognition. Apart from their visual appeal, these algorithms also represent elegant applications of linear algebra and optimization techniques. Topics covered in this course include a recapitulation of relevant linear algebra, introduction to face-recognition, camera calibration, stitched panoramas, edge and blob visual features, structure from motion, color-spaces, segmentation, and an introduction to object-recognition.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students should be able
• describe image formation and camera models;
• calibrate cameras;
• compute image histograms, and basic image processing;
• discriminate among visual features (e.g., corner, edge, blob);
• Properly use computer vision libraries;
• implement computer vision applications.</td></tr>
<tr><td colspan="5">Indicative Literature

D.A. Forsyth and J. Ponce, Computer Vision: A Modern Approach. 2nd edition, 2011.

R. Szeliski, Computer Vision: Algorithms and Applications, Springer, http://szeliski.org/Book, 2010.

Ma et al., An Invitation to 3 D Vision: From Images to Geometric Models, Springer, 2004.</td></tr>
<tr><td colspan="5">Usability and Relationship to other Modules</td></tr>
</table>

- Giving the foundation of computer vision, this module is important for RIS project and for advanced specialization courses.
- Mandatory elective for a major in RIS.
- This module serves as a third year Specialization module for CS major students.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min, Weight: 100%

Scope: All intended learning outcomes of the module

Module achievements: 50% if the assignments correctly solved

7.21 Machine Learning

<table>
<tr><td colspan="3">Module Name
Machine Learning</td><td>Module Code
CO-541</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CO-541-A | Machine Learning | Lecture | 5</td></tr>
<tr><td>Module Coordinator

P. Zaspel</td><td colspan="3">Program Affiliation

• Robotics and Intelligent Systems (RIS)</td><td colspan="2">Mandatory Status

Mandatory for RIS
Mandatory elective for CS</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒None

Knowledge, Abilities, or Skills
• Knowledge and command of probability theory and methods, as in the module “Probability and Random Process (JTMS-12)</td><td>Frequency

Annually (Spring)</td><td colspan="2">Forms of Learning and Teaching

• Class attendance (35 hours)
• Private study (70 hours)
• Exam preparation (20 hours)</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

None</td></tr>
<tr><td colspan="6">Content and Educational Aims

Machine learning (ML) concerns algorithms that are fed with (large quantities of) real-world data, and which return a compressed “model” of the data. An example is the “world model” of a robot; the input data are sensor data streams, from which the robot learns a model of its environment, which is needed, for instance, for navigation. Another example is a spoken language model; the input data are speech recordings, from which ML methods build a model of spoken English; this is useful, for instance, in automated speech recognition systems. There exist many formalisms in which such models can be cast, and an equally large diversity of learning algorithms. However, there is a relatively small number of fundamental challenges that are common to all of these formalisms and algorithms. The lectures introduce such fundamental concepts and illustrate them with a choice of elementary model formalisms (linear classifiers and regressors, radial basis function networks, clustering, online adaptive filters, neural networks, or hidden Markov models). Furthermore, the lectures also (re-)introduce required mathematical material from probability theory and linear algebra.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students should be able to

• understand the notion of probability spaces and random variables;
• understand basic linear modeling and estimation techniques;
• understand the fundamental nature of the “curse of dimensionality;”
• understand the fundamental nature of the bias-variance problem and standard coping strategies;
• use elementary classification learning methods (linear discrimination, radial basis function networks, multilayer perceptrons);
• implement an end-to-end learning suite, including feature extraction and objective function optimization with regularization based on cross-validation.</td></tr>
</table>

Indicative Literature

T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edition, Springer, 2008.

S. Shalev-Shwartz, Shai Ben-David: Understanding Machine Learning, Cambridge University Press, 2014.

C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

T.M. Mitchell, Machine Learning, Mc Graw Hill India, 2017.

Usability and Relationship to other Modules

- Mandatory for a major in RIS
- Mandatory for a minor in RIS
- This module serves as a third Year Specialization module for CS major students.
- This module gives a thorough introduction to the basics of machine learning. It complements the Artificial Intelligence module.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.22 Digital Design

<table>
<tr><td colspan="3">Module Name
Digital Design</td><td>Module Code
CA-S-ECE-803</td><td>Level (type)
Year 3
(CAREER - Specialization)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CA-ECE-803 | Digital Design | Lecture/Lab | 5</td></tr>
<tr><td colspan="2">Module Coordinator

Fangning Hu</td><td colspan="4">Program Affiliation

• Electrical and Computer Engineering (ECE)</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites | Co-requisites | Knowledge, Abilities, or Skills
☒ None | ☒ None</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Mandatory Status

Mandatory elective for ECE, RIS, CS

Forms of Learning and Teaching

• Lecture/Lab (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Students may prepare themselves with books like “Brent E. Nelson, Designing Digital Systems, 2005” and “Pong P. Chu, RTL Hardware Design Using VHDL, A John Wiley & Sons, Inc, Publication, 2006”</td></tr>
<tr><td colspan="6">Content and Educational Aims

The current trend of digital system design is towards hardware description languages (HDLs) that allow compact description of very complex hardware constructs. The module provides a sound introduction to basic components of a digital system such as logic gates, multiplexers, decoders, flip-flops and registers as well as VHDLs such as types, signals, sequential and concurrent statements. Methods and principle of designing complex digital systems such as finite state machines, hierarchical design, pipelined design, RTL design methodology and parameterized design will also be introduced. Students will learn VHDL for programming FPGA boards to realize small digital systems in hardware (i.e. on FPGA boards). Such digital systems could be adders, multiplexers, control units, multipliers, asynchronous serial communication modules (UART). At the end of the module, the students should be able to design a simple digital system by VHDL on an FPGA board.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

By the end of this module, students will be able to
• understand the principle of digital system design based on standard building blocks and components;
• design a complex digital system;
• understand the limitations of a given hardware platform (here FPGAs), modify algorithms where necessary, and structure them suitably in order to optimize performance and complexity;
• use a typical development system;
• program in VHDL;
• program an FPGA board.</td></tr>
</table>

Indicative Literature

Brent E. Nelson, *Designing Digital Systems with SystemVerilog*, 2018, ISBN-13: 978-1980926290

Pong P. Chu, *RTL Hardware Design Using VHDL*, Wiley-IEEE Press, 2006, ISBN-13: 978-0471720928

Usability and Relationship to other Modules

- This module introduces how to design digital systems and how to realize them on a FPGA board which could also serve as a specialization module for students from Computer Science and Robotics and Intelligent Systems.
- Mandatory elective 3rd year Specialization module for ECE, CS and RIS major students.

Examination Type: Module Examination

Assessment Type: written examination	Duration: 120 min
Scope: All intended learning outcomes of the module	Weight: 100%

7.23 Information Theory

<table>
<tr><td colspan="2">Module Name
Information Theory</td><td>Module Code
CO-525</td><td>Level (type)
Year 2 (CORE)</td><td>CP
5.0</td></tr>
<tr><td colspan="5">Module Components</td></tr>
<tr><td>Number</td><td colspan="2">Name</td><td>Type</td><td>CP</td></tr>
<tr><td>CO-525-A</td><td colspan="2">Information Theory</td><td>Lecture</td><td>5.0</td></tr>
<tr><td>Module Coordinator

Prof. Dr.-Ing. Werner Henkel</td><td colspan="2">Program Affiliation

- Electrical and Computer Engineering (ECE)</td><td colspan="2">Mandatory Status

Mandatory for ECE
Mandatory elective for CS and RIS</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites

☒ None</td><td>Co-requisites

☒ None</td><td>Knowledge, Abilities, or Skills

• Signals and Systems contents, such as DFT and convolution
• Notion of probability, combinatorics basics as taught in Methods module “Probability and Random Processes"</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Lectures (35 hours)
• Private Study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Some basic knowledge of communications and sound understanding of probability is recommended. Hence, it is strongly advised to take the methods and skills course Probability and Random Processes prior to this module. Nevertheless, probability basics will also be revised within the module.</td></tr>
<tr><td colspan="5">Content and Educational Aims

Information theory serves as the most important foundation for communication systems. The module provides an analytical framework for modeling and evaluating point-to-point and multi-point communication. After a short rehearsal of probability and random variables and some excursion to random number generation, the key concept of information content of a signal source and information capacity of a transmission medium are precisely defined, and their relationships to data compression algorithms and error control codes are examined in detail. The module aims to install an appreciation for the fundamental capabilities and limitations of information transmission schemes and to provide the mathematical tools for applying these ideas to a broad class of communications systems.
The module contains also a coverage of different source-coding algorithms like Huffman, Lempel-Ziv-(Welch), Shannon-Fano-Elias, Arithmetic Coding, Runlength Encoding, Move-to-Front transform, PPM, and Context Tree Weighting. In Channel coding, finite fields, some basic block and convolutional codes, and the concept of iterative decoding will be introduced. Aside from source and channel aspects, an introduction to security is given, including public-key cryptography. Information theory is a standard module in every communications-oriented Bachelor’s program.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this module, students should be able to</td></tr>
</table>

- explain what is understood as the information content of data and the corresponding limits of data compression algorithms;
- design and apply fundamental algorithms in data compression;
- explain the information theoretic limits of data transmission;
- apply the mathematical basics of channel coding and cryptography;
- implement some channel coding schemes;
- differentiate the principles of encryption and authentication schemes and implement discussed procedures;

Indicative Literature

Thomas M. Cover, Joy A. Thomas, *Elements of Information Theory*, 2nd ed., Wiley, Sept. 2006.

David Salomon, *Data Compression, The Complete Reference*, 4th ed., Springer, 2007.

Usability and Relationship to other Modules

- Although not a mandatory prerequisite, this module is ideally taken before Coding Theory (CA-ECE-802)
- All communications-related modules are naturally based on information theory
- Students from Computer Science or related programs, also students taking Bio-informatics modules, profit from information-theoretic knowledge and source coding (compression) algorithms. Students from Computer Science would also be interested in the algebraic basics for error-correcting codes and cryptology, fields which area also introduced shortly.
- Mandatory for a major in ECE.
- Serves as a mandatory elective 3rd year Specialization module for CS and RIS major students.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of the module.

7.24 Internship / Startup and Career Skills

<table>
<tr><td colspan="2">Module Name
Internship / Startup and Career Skills</td><td>Module Code
CA-INT-900</td><td>Level (type)
Year 3 (CAREER)</td><td>CP
15</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
CA-INT-900-0 | Internship | Internship | 15</td></tr>
<tr><td>Module Coordinator

Predrag Tapavicki & Christin Klähn (CSC Organization); SPC / Faculty Startup Coordinator (Academic responsibility);</td><td colspan="2">Program Affiliation

• CAREER module for undergraduate study programs</td><td colspan="2">Mandatory Status

Mandatory for all undergraduate study programs except IEM</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ at least 15 CP from CORE modules in the major

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• Information provided on CSC pages (see below)
• Major specific knowledge and skills</td><td>Frequency

Annually (Spring/Fall)

Duration
1 semester</td><td colspan="2">Forms of Learning and Teaching

• Internship/Start-up
• Internship event
• Seminars, info-sessions, workshops and career events
• Self-study, readings, online tutorials

Workload

375 Hours consisting of:
• Internship (308 hours)
• Workshops (33 hours)
• Internship Event (2 hours)
• Self-study (32 hours)</td></tr>
<tr><td colspan="5">Recommendations for Preparation

• Reading the information in the menu sections titled “Internship Information,” “Career Events,” “Create Your Application,” and “Seminars & Workshops” at the Career Services Center website: https://jacobs-university.jobteaser.com/en/users/sign_in?back_to_after_login=%2F
• Completing all four online tutorials about job market preparation and the application process, which can be found here: https://jacobs-university.jobteaser.com/en/users/sign_in?back_to_after_login=%2F
• Participating in the internship events of earlier classes</td></tr>
<tr><td colspan="5">Content and Educational Aims

The aims of the internship module are reflection, application, orientation, and development: for students to reflect on their interests, knowledge, skills, their role in society, the relevance of their major subject to society, to apply these skills and this knowledge in real life whilst getting practical experience, to find a professional orientation, and to develop their personality and in their career. This module supports the programs’ aims of preparing students for gainful, qualified employment and the development of their personality.
The full-time internship must be related to the students’ major area of study and extends lasts a minimum of two consecutive months, normally scheduled just before the 5th semester, with the internship event and submission of the internship report in the 5th semester. Upon approval by the SPC and CSC, the internship may take place at other times, such as before teaching starts in the 3rd semester or after teaching finishes in the 6th semester. The Study</td></tr>
</table>

Program Coordinator or their faculty delegate approves the intended internship a priori by reviewing the tasks in either the Internship Contract or Internship Confirmation from the respective internship institution or company. Further regulations as set out in the Policies for Bachelor Studies apply.
Students will be gradually prepared for the internship in semesters 1 to 4 through a series of mandatory information sessions, seminars, and career events.
The purpose of the Career Services Information Sessions is to provide all students with basic facts about the job market in general, and especially in Germany and the EU, and services provided by the Career Services Center.
In the Career Skills Seminars, students will learn how to engage in the internship/job search, how to create a competitive application (CV, Cover Letter, etc.), and how to successfully conduct themselves at job interviews and/or assessment centers. In addition to these mandatory sections, students can customize their skill set regarding application challenges and their intended career path in elective seminars.
Finally, during the Career Events organized by the Career Services Center (e.g. the annual Jacobs Career Fair and single employer events on and off campus), students will have the opportunity to apply their acquired job market skills in an actual internship/job search situation and to gain their desired internship in a high-quality environment and with excellent employers.
As an alternative to the full-time internship, students can apply for the StartUp Option. Following the same schedule as the full-time internship, the StartUp Option allows students who are particularly interested in founding their own company to focus on the development of their business plan over a period of two consecutive months. Participation in the StartUp Option depends on a successful presentation of the student's initial StartUp idea. This presentation will be held at the beginning of the 4th semester. A jury of faculty members will judge the student's potential to realize their idea and approve the participation of the students. The StartUp Option is supervised by the Faculty StartUp Coordinator. At the end of StartUp Option, students submit their business plan. Further regulations as outlined in the Policies for Bachelor Studies apply.
The concluding Internship Event will be conducted within each study program (or a cluster of related study programs) and will formally conclude the module by providing students the opportunity to present on their internships and reflect on the lessons learned within their major area of study. The purpose of this event is not only to self-reflect on the whole internship process, but also to create a professional network within the academic community, especially by entering the Alumni Network after graduation. It is recommended that all three classes (years) of the same major are present at this event to enable networking between older and younger students and to create an educational environment for younger students to observe the "lessons learned" from the diverse internships of their elder fellow students.

Intended Learning Outcomes

By the end of this module, students should be able to

- describe the scope and the functions of the employment market and personal career development;
- apply professional, personal, and career-related skills for the modern labor market, including self-organization, initiative and responsibility, communication, intercultural sensitivity, team and leadership skills, etc.;
- independently manage their own career orientation processes by identifying personal interests, selecting appropriate internship locations or start-up opportunities, conducting interviews, succeeding at pitches or assessment centers, negotiating related employment, managing their funding or support conditions (such as salary, contract, funding, supplies, work space, etc.);
- apply specialist skills and knowledge acquired during their studies to solve problems in a professional environment and reflect on their relevance in employment and society;
- justify professional decisions based on theoretical knowledge and academic methods;
- reflect on their professional conduct in the context of the expectations of and consequences for employers and their society;
- reflect on and set their own targets for the further development of their knowledge, skills, interests, and values;
- establish and expand their contacts with potential employers or business partners, and possibly other students and alumni, to build their own professional network to create employment opportunities in the future;
- discuss observations and reflections in a professional network.

Indicative Literature

Not specified

Usability and Relationship to other Modules

- Mandatory for a major in BCCB, CBT, CS, EES, GEM, IBA, IRPH, ISCP, Math, MCCB, Physics, RIS, and SMP.
- This module applies skills and knowledge acquired in previous modules to a professional environment and provides an opportunity to reflect on their relevance in employment and society. It may lead to thesis topics.

Examination Type: Module Examination	
Assessment Type: Internship Report or Business Plan and Reflection	Length: approx. 3.500 words
Scope: All intended learning outcomes	Weight: 100%

7.25 Bachelor Thesis and Seminar

<table>
<tr><td colspan="3">Module Name

Bachelor Thesis and Seminar</td><td>Module Code

CA-CS-800</td><td>Level (type)

Year 3 (CAREER)</td><td>CP

15</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
CA-CS-800-T | Thesis | Thesis | 12
CA-CS-800-S | Thesis Seminar | Seminar | 3</td></tr>
<tr><td>Module Coordinator

Study Program Chair</td><td colspan="3">Program Affiliation

• All undergraduate programs</td><td colspan="2">Mandatory Status

Mandatory for all undergraduate programs</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Students must be in their third year and have taken at least 30 CP from CORE modules in their major.

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• comprehensive knowledge of the subject and deeper insight into the chosen topic;
• ability to plan and undertake work independently;
• skills to identify and critically review literature.</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Self-study/lab work (350 hours)
• Seminars (25 hours)

Workload

375 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

• Identify an area or a topic of interest and discuss this with your prospective supervisor in a timely manner.
• Create a research proposal including a research plan to ensure timely submission.
• Ensure you possess all required technical research skills or are able to acquire them on time.
• Review the University's Code of Academic Integrity and Guidelines to Ensure Good Academic Practice.</td></tr>
</table>

Content and Educational Aims

This module is a mandatory graduation requirement for all undergraduate students to demonstrate their ability to address a problem from their respective major subject independently using academic/scientific methods within a set time frame. Although supervised, this module requires students to be able to work independently and systematically and set their own goals in exchange for the opportunity to explore a topic that excites and interests them personally and that a faculty member is interested in supervising. Within this module, students apply their acquired knowledge about their major discipline and their learned skills and methods for conducting research, ranging from the identification of suitable (short-term) research projects, preparatory literature searches, the realization of discipline-specific research, and the documentation, discussion, interpretation, and communication of research results.

This module consists of two components, an independent thesis and an accompanying seminar. The thesis component must be supervised by a Jacobs University faculty member and requires short-term research work, the results of which must be documented in a comprehensive written thesis including an introduction, a justification of the methods, results, a discussion of the results, and a conclusion. The seminar provides students with the opportunity to practice their ability to present, discuss, and justify their and other students' approaches, methods, and results at various stages of their research in order to improve their academic writing, receive and reflect on formative feedback, and therefore grow personally and professionally.

Intended Learning Outcomes

On completion of this module, students should be able to

1. independently plan and organize advanced learning processes;
2. design and implement appropriate research methods, taking full account of the range of alternative techniques and approaches;
3. collect, assess, and interpret relevant information;
4. draw scientifically-founded conclusions that consider social, scientific, and ethical factors;
5. apply their knowledge and understanding to a context of their choice;
6. develop, formulate, and advance solutions to problems and debates within their subject area, and defend these through argument;
7. discuss information, ideas, problems, and solutions with specialists and non-specialists.

Usability and Relationship to other Modules

- This module builds on all previous modules in the undergraduate program. Students apply the knowledge, skills, and competencies they have acquired and practiced during their studies, including research methods and their ability to acquire additional skills independently as and if required.

Indicative Literature

Justin Zobel, *Writing for Computer Science, 3rd edition,* Springer, 2015.

Examination Type: Module Component Examinations

Module Component 1: Thesis
Assessment type: Thesis
Scope: All intended learning outcomes, mainly 1-6.
Weight: 80%
Length: approx. 6.000 – 8.000 words (15 – 25 pages), excluding front and back matter.

Module Component 2: Seminar
Assessment type: Presentation
Duration: approx. 15 to 30 minutes
Weight: 20%

Scope: The presentation focuses mainly on ILOs 6 and 7, but by nature of these ILOs it also touches on the others.

Completion: To pass this module, the examination of each module component has to be passed with at least 45%.

Two separate assessments are justified by the size of this module and the fact that the justification of solutions to problems and arguments (ILO 6) and discussion (ILO 7) should at least have verbal elements. The weights of the types of assessments are commensurate with the sizes of the respective module components.

7.26 Jacobs Track Modules

7.26.1 Methods and Skills Modules

7.26.1.1 Calculus and Elements of Linear Algebra I

<table>
<tr><td colspan="2">Module Name
Calculus and Elements of Linear Algebra I</td><td>Module Code
JTMS-MAT-09</td><td>Level (type)
Year 1 (Methods)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
JTMS-09 | Calculus and Elements of Linear Algebra I | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Marcel Oliver, Tobias Preußer</td><td colspan="2">Program Affiliation

• Jacobs Track – Methods and Skills</td><td colspan="2">Mandatory Status

Mandatory for CS, ECE, RIS, MATH and Physics
Mandatory elective for EES</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• Knowledge of Pre-Calculus at High School level (Functions, inverse functions, sets, real numbers, polynomials, rational functions, trigonometric functions, logarithm and exponential function, parametric equations, tangent lines, graphs, elementary methods for solving systems of linear and nonlinear equations)
• Knowledge of Analytic Geometry at High School level (vectors, lines, planes, reflection, rotation, translation, dot product, cross product, normal vector, polar coordinates)
• Some familiarity with elementary Calculus (limits, derivative) is helpful, but not strictly required.</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Review all of higher-level High School Mathematics, in particular the topics explicitly named in “Entry Requirements – Knowledge, Ability, or Skills” above.</td></tr>
<tr><td colspan="5">Content and Educational Aims

This module is the first in a sequence introducing mathematical methods at the university level in a form relevant for study and research in the quantitative natural sciences, engineering, Computer Science, and Mathematics.</td></tr>
</table>

The emphasis in these modules is on training operational skills and recognizing mathematical structures in a problem context. Mathematical rigor is used where appropriate. However, a full axiomatic treatment of the subject is provided in the first-year modules "Analysis I" and "Linear Algebra".

The lecture comprises the following topics

- Brief review of number systems, elementary functions, and their graphs
- Brief introduction to complex numbers
- Limits for sequences and functions
- Continuity
- Derivatives
- Curve sketching and applications (isoperimetric problems, optimization, error propagation)
- Introduction to Integration and the Fundamental Theorem of Calculus
- Review of elementary analytic geometry
- Vector spaces, linear independence, bases, coordinates
- Matrices and matrix algebra
- Solving linear systems by Gauss elimination, structure of general solution
- Matrix inverse

Intended Learning Outcomes

By the end of the module, students will be able to

- apply the methods described in the content section of this module description to the extent that they can solve standard text-book problems reliably and with confidence;
- recognize the mathematical structures in an unfamiliar context and translate them into a mathematical problem statement;
- recognize common mathematical terminology used in textbooks and research papers in the quantitative sciences, engineering, and mathematics to the extent that they fall into the content categories covered in this module.

Indicative Literature

S.I. Grossman (2014). Calculus of one variable, 2nd edition. Cambridge: Academic Press.
S.A. Leduc (2003). Linear Algebra. Hoboken: Wiley.
K. Riley, M. Hobson, S. Bence (2006). Mathematical Methods for Physics and Engineering, third edition. Cambridge: Cambridge University Press.

Usability and Relationship to other Modules

- The module is a mandatory / mandatory elective module of the Methods and Skills area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- The module is followed by "Calculus and Elements of Linear Algebra II". All students taking this module are expected to register for the follow-up module.
- A rigorous treatment of Calculus is provided in the module "Analysis I". All students taking "Analysis I" are expected to either take this module or exceptionally satisfy the conditions for advanced placement as laid out in the Jacobs Academic Policies for Undergraduate Study.
- The second-semester module "Linear Algebra" will provide a complete proof-driven development of the theory of Linear Algebra. Students enrolling in "Linear Algebra" are expected to have taken this module; in particular, the module "Linear Algebra" will assume that students are proficient in the operational aspects of Gauss elimination, matrix inversion, and their elementary applications.
- This module is a prerequisite for the module "Applied Mathematics" which develops more advanced theoretical and practical mathematical tools essential for any physicist or mathematician.
- Mandatory for a major in CS, ECE, RIS, MATH and Physics
- Mandatory elective for a major in EES.
- Pre-requisite for Calculus and Elements of Linear Algebra II
- Elective for all other study programs.

Examination Type: Module Examination

Assessment type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of this module

7.26.1.2 Calculus and Elements of Linear Algebra II

<table>
<tr><td colspan="3">Module Name
Calculus and Elements of Linear Algebra II</td><td>Module Code
JTMS-MAT-10</td><td>Level (type)
Year 1 (Methods)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
JTMS-10 | Calculus and Elements of Linear Algebra II | Lecture | 5</td></tr>
<tr><td colspan="2">Module Coordinator

Marcel Oliver, Tobias Preußer</td><td colspan="2">Program Affiliation

• Jacobs Track – Methods and Skills</td><td colspan="2">Mandatory Status

Mandatory for CS, ECE, MATH, Physics, RIS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ Calculus and Elements of Linear Algebra I

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• None beyond formal pre-requisites</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Review the content of Calculus and Elements of Linear Algebra I</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module is the second in a sequence introducing mathematical methods at the university level in a form relevant for study and research in the quantitative natural sciences, engineering, Computer Science, and Mathematics. The emphasis in these modules is on training operational skills and recognizing mathematical structures in a problem context. Mathematical rigor is used where appropriate. However, a full axiomatic treatment of the subject is provided in the first-year modules “Analysis I” and “Linear Algebra”.

The lecture comprises the following topics
• Directional derivatives, partial derivatives
• Linear maps
• The total derivative as a linear map
• Gradient and curl (elementary treatment only, for more advanced topics, in particular the connection to the Gauss and Stokes’ integral theorems, see module “Applied Mathematics”
• Optimization in several variables, Lagrange multipliers
• Elementary ordinary differential equations
• Eigenvalues and eigenvectors
• Hermitian and skew-Hermitian matrices
• First important example of eigendecompositions: Linear constant-coefficient ordinary differential equations
• Second important example of eigendecompositions: Fourier series
• Fourier integral transform
• Matrix factorizations: Singular value decomposition with applications, LU decomposition, QR decomposition</td></tr>
<tr><td colspan="6">Intended Learning Outcomes
By the end of the module, students will be able to
• apply the methods described in the content section of this module description to the extent that they can solve standard text-book problems reliably and with confidence;
• recognize the mathematical structures in an unfamiliar context and translate them into a mathematical problem statement;</td></tr>
</table>

- recognize common mathematical terminology used in textbooks and research papers in the quantitative sciences, engineering, and mathematics to the extent that they fall into the content categories covered in this module.

Indicative Literature

S.I. Grossman (2014). Calculus of one variable, 2nd edition. Cambridge: Academic Press.
S.A. Leduc (2003). Linear Algebra. Hoboken: Wiley.
K. Riley, M. Hobson, S. Bence (2006). Mathematical Methods for Physics and Engineering, third edition. Cambridge: Cambridge University Press.

Usability and Relationship to other Modules

- The module is a mandatory / mandatory elective module of the Methods and Skills area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- A more advanced treatment of multi-variable Calculus, in particular, its applications in Physics and Mathematics, is provided in the second-semester module "Applied Mathematics". All students taking "Applied Mathematics" are expected to take this module as well as the module topics are closely synchronized.
- The second-semester module "Linear Algebra" provides a complete proof-driven development of the theory of Linear Algebra. Diagonalization is covered more abstractly, with particular emphasis on degenerate cases. The Jordan normal form is also covered in "Linear Algebra", not in this module.
- Mandatory for CS, ECE, MATH, Physics and RIS.
- Elective for all other study programs.

Examination Type: Module Examination

Assessment type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of this module

7.26.1.3 Probability and Random Processes

<table>
<tr><td colspan="3">Module Name
Probability and Random Processes</td><td>Module Code
JTMS-MAT-12</td><td>Level (type)
Year 2 (Methods)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
JTMS-12 | Probability and random processes | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Marcel Oliver, Tobias Preußer</td><td colspan="3">Program Affiliation

• Jacobs Track – Methods and Skills</td><td colspan="2">Mandatory Status

Mandatory for CS, ECE, MATH, Physics, RIS
Mandatory elective for EES</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites

☒
Calculus and Elements of Linear Algebra I & II

Co-requisites

None

Knowledge, Abilities, or Skills

• Knowledge of calculus at the level of a first year calculus module (differentiation, integration with one and several variables, trigonometric functions, logarithms and exponential functions).
• Knowledge of linear algebra at the level of a first year university module (eigenvalues and eigenvectors, diagonalization of matrices).
• Some familiarity with elementary probability theory at the high school level.</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Review all of the first year calculus and linear algebra modules as indicated in “Entry Requirements – Knowledge, Ability, or Skills” above.</td></tr>
</table>

Content and Educational Aims

This module aims to provide a basic knowledge of probability theory and random processes suitable for students in engineering, Computer Science, and Mathematics. The module provides students with basic skills needed for formulating real-world problems dealing with randomness and probability in mathematical language, and methods for applying a toolkit to solve these problems. Mathematical rigor is used where appropriate. A more advanced treatment of the subject is deferred to the third-year module *Stochastic Processes*.

The lecture comprises the following topics

- Brief review of number systems, elementary functions, and their graphs
- Outcomes, events and sample space.
- Combinatorial probability.
- Conditional probability and Bayes' formula.
- Binomials and Poisson-Approximation
- Random Variables, distribution and density functions.
- Independence of random variables.
- Conditional Distributions and Densities.
- Transformation of random variables.
- Joint distribution of random variables and their transformations.
- Expected Values and Moments, Covariance.
- High dimensional probability: Chebyshev and Chernoff bounds.
- Moment-Generating Functions and Characteristic Functions,
- The Central limit theorem.
- Random Vectors and Moments, Covariance matrix, Decorrelation.
- Multivariate normal distribution.
- Markov chains, stationary distributions.

Intended Learning Outcomes

By the end of the module, students will be able to

- command the methods described in the content section of this module description to the extent that they can solve standard text-book problems reliably and with confidence;
- recognize the probabilistic structures in an unfamiliar context and translate them into a mathematical problem statement;
- recognize common mathematical terminology used in textbooks and research papers in the quantitative sciences, engineering, and mathematics to the extent that they fall into the content categories covered in this module.

Indicative Literature

J. Hwang and J.K. Blitzstein (2019). Introduction to Probability, second edition. London: Chapman & Hall.

S. Ghahramani. Fundamentals of Probability with Stochastic Processes, fourth edition. Upper Saddle River: Prentice Hall.

Usability and Relationship to other Modules

- The module is a mandatory / mandatory elective module of the Methods and Skills area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students taking this module are expected to be familiar with basic tools from calculus and linear algebra.
- Mandatory for a major in CS, ECE, MATH, Physics and RIS.
- Mandatory elective for a major in EES (if pre-requisites are met).
- Elective for all other study programs.

Examination Type: Module Examination

Assessment type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of this module

7.26.1.4 Numerical Methods

<table>
<tr><td colspan="3">Module Name
Numerical Methods</td><td>Module Code
JTMS-MAT-13</td><td>Level (type)
Year 2 (Methods)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
JTMS-13 | Numerical Methods | Lecture | 5</td></tr>
<tr><td>Module Coordinator

Marcel Oliver, Tobias Preußer</td><td colspan="3">Program Affiliation

• Jacobs Track – Methods and Skills</td><td colspan="2">Mandatory Status

Mandatory for ECE, MATH, Physics
Mandatory elective for CS and RIS</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• Knowledge of Calculus (functions, inverse functions, sets, real numbers, sequences and limits, polynomials, rational functions, trigonometric functions, logarithm and exponential function, parametric equations, tangent lines, graphs, derivatives, anti-derivatives, elementary techniques for solving equations)
• Knowledge of Linear Algebra (vectors, matrices, lines, planes, n-dimensional Euclidean vector space, rotation, translation, dot product (scalar product), cross product, normal vector, eigenvalues,</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
</table>

	eigenvectors, elementary techniques for solving systems of linear equations)		

Recommendations for Preparation

Taking Calculus and Elements of Linear Algebra II before taking this module is recommended, but not required. A thorough review of Calculus and Elements of Linear Algebra, with emphasis on the topics listed as "Knowledge, Abilities, or Skills" is recommended.

Content and Educational Aims

This module covers calculus-based numerical methods, in particular root finding, interpolation, approximation, numerical differentiation, numerical integration (quadrature), and a first introduction to the numerical solution of differential equations.

The lecture comprises the following topics

- number representations
- Gaussian elimination
- LU decomposition
- Cholesky decomposition
- iterative methods
- bisection method
- Newton's method
- secant method
- polynomial interpolation
- Aitken's algorithm
- Lagrange interpolation
- Newton interpolation
- Hermite interpolation
- Bezier curves
- De Casteljau's algorithm
- piecewise interpolation
- Spline interpolation
- B-Splines
- Least-squares approximation
- polynomial regression
- difference schemes
- Richardson extrapolation
- Quadrature rules
- Monte Carlo integration
- time stepping schemes for ordinary differential equations
- Runge Kutta schemes
- finite difference method for partial differential equations

Intended Learning Outcomes

By the end of the module, students will be able to

- describe the basic principles of discretization used in the numerical treatment of continuous problems;
- command the methods described in the content section of this module description to the extent that they can solve standard text-book problems reliably and with confidence;
- recognize mathematical terminology used in textbooks and research papers on numerical methods in the quantitative sciences, engineering, and mathematics to the extent that they fall into the content categories covered in this module;
- implement simple numerical algorithms in a high-level programming language;
- understand the documentation of standard numerical library code and understand the potential limitations and caveats of such algorithms.

Indicative Literature

D. Kincaid and W. Cheney (1991). Numerical Analysis: Mathematics of Scientific Computing. Pacific Grove: Brooks/Cole Publishing.

W. Boehm and H. Prautzsch (1993). Numerical Methods. Natick: AK Peters.

Usability and Relationship to other Modules

- The module is a mandatory / mandatory elective module of the Methods and Skills area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- This module is a co-recommendation for the module "Applied Dynamical Systems Lab", in which the actual implementation in a high-level programming language of the learned methods will be covered.
- Mandatory for a major in ECE, MATH, and Physics.
- Mandatory elective for a major in CS and RIS.
- Elective for all other study programs.

Examination Type: Module Examination

Assessment type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of this module.

7.26.1.5 Discrete Mathematics

<table>
<tr><td colspan="3">Module Name

Discrete Mathematics</td><td>Module Code

CO-501</td><td>Level (type)

Year 2/3 (CORE)</td><td>CP

5.0</td></tr>
<tr><td colspan="6">Module Components

<table><tr><td>Number</td><td>Name</td><td>Type</td><td>CP</td></tr><tr><td>CO-501-A</td><td>Discrete Mathematics</td><td>Lecture</td><td>5.0</td></tr></table></td></tr>
<tr><td colspan="2">Module Coordinator
K. Mallahi-Karai</td><td colspan="2">Program Affiliation

• Mathematics</td><td colspan="2">Mandatory Status

Mandatory elective for Mathematics, CS, Physics and RIS</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

<table><tr><td>Pre-requisites</td><td>Co-requisites</td><td>Knowledge, Abilities, or Skills</td></tr><tr><td>☒ None</td><td>☒ None</td><td>• Basic university mathematics: can be acquired via the Methods Modules “Calculus and Elements of Linear Algebra I + II” or “Applied Calculus” and “Finite Mathematics”</td></tr></table></td><td>Frequency

Annually (Spring)</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private Study (90 hours)</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation
• Some basic familiarity with linear algebra is useful, but not technically required.
• It is recommended to have taken the Methods module: Calculus and Elements of Linear Algebra I + II</td></tr>
<tr><td colspan="6">Content and Educational Aims

This module is an introductory lecture in discrete mathematics. The lecture consists of two main components, enumerative combinatorics and graph theory. The lecutre emphasizes connections of discrete mathematics with other areas of mathematics such as linear algebra and basic probability, and outlines applications to areas of computer science, cryptography, etc. where employment of ideas from discrete mathematics has proven to be fruitful. The first part of the lecture—enumerative combinatorics—deals with several classical enumeration problems (Binomial coefficients, Stirling numbers), counting under group actions and generating function. The second half of the lecture—graph theory—includes a discussion of basic notions such as chromatic number, planarity, matchings in graphs, Ramsey theory, and expanders, and their applications.</td></tr>
</table>

Intended Learning Outcomes

By the end of the module, students will be able to

- demonstrate their mastery of basic tools in discrete mathematics.
- develop the ability to use discrete mathematics concepts (such as graphs) to model problems in computer science.
- analyze the definition of basic combinatorial objects such as graphs, permutations, partitions, etc.
- formulate and design methods sand algorithms for solving applied problems basic on concepts from discrete mathematics.

Indicative Literature

J.H. van Lint and R.M. Wilson (2001). A Course in Combinatorics, second edition. Cambridge: Cambridge University Press.

B. Bollobas (1998). Modern Graph Theory, Berlin: Springer.

Usability and Relationship to other Modules

- This module is a specialization / CORE module in Mathematics to be taken in Semester 4 or 6.
- This module is recommended for students pursuing a minor in Mathematics
- This module serves as a mandatory elective Methods and Skills module for CS, Physics and RIS
- This module is a good option as an elective module for students in RIS.

Examination Type: Module Examination

Assessment Type: Written examination

Duration: 120 min
Weight: 100%

Scope: All intended learning outcomes of this module

7.26.2 Big Questions Modules

7.26.2.1 Water: The Most Precious Substance on Earth

<table>
<tr><td colspan="2">Module Name
Water: The Most Precious Substance on Earth</td><td>Module Code
JTBQ-BQ-002</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
JTBQ-002 | Water: The Most Precious Substance on Earth | Lecture/Tutorial | 5</td></tr>
<tr><td>Module Coordinator

M. Bau and D. Mosbach</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites: ☒ None
Co-requisites: ☒ None
Knowledge, Abilities, or Skills:
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (part I: Fall; part II: Spring)

Duration

2 semesters</td><td>Forms of Learning and Teaching

• Lectures (17.5 hours)
• Project work (90 hours)
• Private study (17.5 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage on the module's topics in question.</td></tr>
</table>

Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizons with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
Water is the basic prerequisite for life on our planet, but it has become a scarce resource and a valuable commodity. Water is of fundamental importance to the world's economy and global food supply, in addition to being a driving force behind geopolitical conflict. In this module, the profound impact of water on all aspects of human life will be addressed from very different perspectives: from the natural and environmental sciences and engineering, and from the social and cultural sciences.
Following topical lectures in the Fall semester, students will work on projects on the occasion of the World Water Day (March 22) in small teams comprised of students from various disciplines and with different cultural backgrounds. This teamwork will be accompanied by related tutorials.

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- advance a knowledge-based opinion on the complex module topics: on the physio-chemical properties of water, its origin and history, on the importance of water as a resource, on physical and economic freshwater scarcity, on the risks of water pollution and the challenges faced by waste water treatment, on the concept of virtual water, on the bottled water industry, and on the cultural values and meanings of water;
- formulate coherent written and oral contributions (e.g., to panel discussions) on the topic;
- perform well-organized teamwork;
- present a self-designed project in a university-wide context.

Indicative Literature

Finney, John (2015). Water. A Very Short Introduction. Oxford: Oxford University Press.

Zetland, David (2011). The End of Abundance: Economic Solutions to Water Scarcity. California: Aguanomics Press.

United Nation (January 2016): Sustainable Development Goals. Retrieved from https://www.ipcc.ch

Usability and Relationship to other Modules

- This module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Component 1: Written examination — Duration: 60 min, Weight: 50%

Assessment Component 2: Team project — Weight: 50%

Scope: All intended learning outcomes of the module

Completion: This module is passed with an assessment-component weighted average grade of 45% or higher.

7.26.2.2 Ethics in Science and Technology

<table>
<tr><td colspan="2">Module Name
Ethics in Science and Technology</td><td>Module Code
JTBQ-BQ-003</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
5.0</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
JTBQ-003 | Ethics in Science and Technology | Lecture /Projects | 5.0</td></tr>
<tr><td>Module Coordinator

A. Lerchl</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status

• Mandatory for CBT
• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites | Co-requisites | Knowledge, Abilities, or Skills
☒ None | ☒ None | • The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Each semester (Fall & Spring)</td><td>Forms of Learning and Teaching

• Lectures (35 hours)
• Project work (55 hours)
• Private study (35 hours)</td></tr>
<tr><td>Duration

1 semester</td><td>Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage of the scientific topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All “Big Questions” (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students’ horizons with applied problem solving that extends beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
Ethics is an often neglected, yet essential part of science and technology. Our decisions about right and wrong influence the way in which our inventions and developments change the world. A wide array of examples will be presented and discussed, e.g., the foundation of ethics, individual vs. population ethics, artificial life, stem cells, animal rights, abortion, pre-implantation diagnostics, legal and illegal drugs, the pharmaceutical industry, gene modification, clinical trials and research with test persons, weapons of mass destruction, data fabrication, and scientific fraud.</td></tr>
</table>

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- summarize and explain ethical principles;
- critically look at scientific results that seem too good to be true;
- apply the ethical concepts to virtually all areas of science and technology;
- discover the responsibilities of society and of the individual for ethical standards;
- understand and judge the ethical dilemmas in many areas of the daily life;
- discuss the ethics of gene modification at the level of cells and organisms;
- reflect on and evaluate clinical trials in relation to the Helsinki Declaration;
- distinguish and evaluate the ethical guidelines for studies with test persons;
- complete a self-designed project;
- overcome general teamwork problems;
- perform well-organized project work.

Indicative Literature

Not specified.

Usability and Relationship to other Modules

- Mandatory for CBT
- This module is a mandatory elective module in the Big Questions area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Component 1: Written examination	Duration: 60 min
	Weight: 50%
Assessment Component 2: Team project	Weight: 50%

Scope: All intended learning outcomes of the module

Completion: This module is passed with an assessment-component weighted average grade of 45% or higher.

7.26.2.3 Global Health – Historical context and future challenges

<table>
<tr><td colspan="2">Module Name
Global Health – Historical context and future challenges</td><td>Module Code
JTBQ-BQ-004</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
JTBQ-004 | Global Health – Historical context and future challenges | Lecture | 5</td></tr>
<tr><td>Module Coordinator

A. M. Lisewski</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td colspan="2">Forms of Learning and Teaching

• Lectures (35 hours)
• Private study (90 hours)

Workload

125 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage on the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizons with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
This module gives a historical, societal, technical, scientific, and medical overview of the past and future milestones and challenges of global health. Particular focus is put on future global health issues in a world that is interconnected both through mobility and communication networks. This module presents the main milestones along the path to modern health systems, including the development of public hygiene, health monitoring and disease response, and health-related breakthroughs in science, technology, and the economy. Focus is given to pediatric, maternal, and adolescent health, as these are the areas most critical to the well-being of future generations. This module also provides key concepts in global health, epidemiology, and demographics, such as the connection between a society's economic level and its population's health status, measures of health status, demographic and epidemiologic transitions, and modern issues such as the growing fragmentation (at a personal level) of disease conditions and the resulting emergence of personalized medicine. Finally, attention is also given to less publicly prominent global health issues, such as re-emerging diseases, neglected tropical diseases, and complex humanitarian crises.</td></tr>
</table>

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- explain the historical context of current global health surveillance, response systems, and institutions;
- discuss and evaluate the imminent and future challenges to public hygiene and response to disease outbreaks in the context of a global societal network.

Indicative Literature

Richard Skolnik (2015). Global Health 101 (Essential Public Health). Burlington: Jones and Bartlett Publishers, Inc.

Usability and Relationship to other Modules

- The module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Written examination	Duration: 60 min.
Scope: All intended learning outcomes of the module	Weight: 100%

7.26.2.4 Global Existential Risks

<table>
<tr><td colspan="2">Module Name
Global Existential Risks</td><td>Module Code
JTBQ-BQ-005</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components

Number | Name | Type | CP
JTBQ-005 | Global Existential Risks | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator

M. A. Lisewski</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (Spring)</td><td>Forms of Learning and Teaching

• Lectures (17.5 hours)
• Private study (45 hours)</td></tr>
<tr><td>Duration

1 semester</td><td>Workload

62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage on the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizons with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
The more we develop science and technology, the more we also learn about catastrophic and, in the worst case, even existential global dangers that put the entire human civilization at risk of collapse. These doomsday scenarios therefore directly challenge humanity's journey through time as an overall continuous and sustainable process that progressively leads to a more complex but still largely stable human society. The module presents the main known varieties of existential risks, including, for example, astrophysical, planetary, biological, and technological events or critical transitions that have the capacity to severely damage or even eradicate earth-based human civilization as we know it. Furthermore, this module offers a description of the characteristic features of these risks in comparison to more conventional risks, such as natural disasters, and a classification of global existential risks based on parameters such as range, intensity, probability of occurrence, and imminence. Finally, this module reviews several hypothetical monitoring and early warning systems as well as analysis methods that could potentially be used in strategies, if not to eliminate, then at least to better understand and ideally to minimize</td></tr>
</table>

imminent global existential risks. This interdisciplinary module will allow students to explore this topic across diverse subject fields.

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

- use their factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- explain the varieties of global existential risks;
- discuss approaches to minimize these risks;
- formulate coherent written and oral contributions on this topic.

Indicative Literature

Nick Bostrom, Milan M. Cirkovic (eds.) (2011). Global Catastrophic Risk.Oxford: Oxford University Press.

Murray Shanahan (2015). The Technological Singularity. Cambridge:The MIT Press.

Martin Rees (2003) Our Final Hour. New York: Basic Books.

Usability and Relationship to other Modules

- This module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Written examination	Duration: 60 min.
Scope: All intended learning outcomes of the module	Weight: 100%

7.26.2.5 Future: From Predictions and Visions to Preparations and Actions

<table>
<tr><td colspan="2">Module Name
Future: From Predictions and Visions to Preparations and Actions</td><td>Module Code
JTBQ-BQ-006</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components
Number | Name | Type | CP
JTBQ-006 | Future: From Predictions and Visions to Preparations and Actions | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator

Joachim Vogt</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (Spring)</td><td>Forms of Learning and Teaching

• Lecture (17.5 hours)
• Private study (45 hours)</td></tr>
<tr><td>Duration

1 semester</td><td>Workload

62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage of the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizons with applied problem solving that extend beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
This module addresses selected topics related to the future as a general concept in science, technology, culture, literature, ecology, and economy, and it consists of three parts. The first part (Future Continuous) discusses forecasting methodologies rooted in the idea that key past and present processes are understood and continue to operate such that future developments can be predicted. General concepts covered in this context include determinism, uncertainty, evolution, and risk. Mathematical aspects of forecasting are also discussed. The second part (Future Perfect) deals with human visions of the future as reflected in the arts and literature, ranging from ideas of utopian societies and technological optimism to dystopian visions in science fiction. The third part (Future Now) concentrates on important current developments—such as trends in technology, scientific breakthroughs, the evolution of the Earth system, and climate change—and concludes with opportunities and challenges for present and future generations.</td></tr>
</table>

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, student should be able to

- use their factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- distinguish and qualify important approaches to forecasting and prediction;
- summarize the history of utopias, dystopias, and the ideas presented in classical science fiction;
- characterize current developments in technology, ecology, society, and their implications for the future.

Indicative Literature

United Nations (2015, September) Millennium Development Goals. Retrieved from http://www.un.org/millenniumgoals.
United Nation (2016, January): Sustainable Development Goals. Retrieved from http://catalog.jacobs-university.de/search~S0
United Nations University. https://unu.edu
US National Intelligence Council (2017). Global Trends. Retrieved from https://www.dni.gov/index.php/global-trends-home.
International Panel on Climate Change. Retrieved from https://www.ipcc.ch.
World Inequality Lab (2017, December). World Inequality Report 2018. Retrieved from https://wir2018.wid.world.
World Health Organization. Retrieved from http://www.who.int.
World Trade Organization. Retrieved from *https://www.wto.org*
Gapminder. Retrieved from https://www.gapminder.org.
World Bank. Retrieved from http://www.worldbank.org.

Usability and Relationship to other Modules

- This module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 60 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.26.2.6 Climate Change

<table>
<tr><td colspan="2">Module Name
Climate Change</td><td>Module Code
JTBQ-BQ-007</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components
Number | Name | Type | CP
JTBQ-007 | Climate Change | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator
L. Thomsen/
V. Unnithan</td><td colspan="2">Program Affiliation
• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status
• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements
Pre-requisites: ☒ None
Co-requisites: ☒ None
Knowledge, Abilities, or Skills:
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency
Annually (Spring)</td><td>Forms of Learning and Teaching
• Lecture (17.5 hours)
• Private study (45 hours)</td></tr>
<tr><td>Duration
1 semester</td><td>Workload
62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation
Critically following media coverage of the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims
All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizon with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
This module will give a brief introduction into the development of the atmosphere throughout Earth's history from the beginning of the geological record up to modern times, and will focus on geological, cosmogenic, and anthropogenic changes. Several major events in the evolution of the Earth that had a major impact on climate will be discussed, such as the evolution of an oxic atmosphere and ocean, the onset of early life, snowball Earth, and modern glaciation cycles. In the second part, the module will focus on the human impact on present climate change and global warming. Causes and consequences, including case studies and methods for studying climate change, will be presented and possibilities for climate mitigation (geo-engineering) and adapting our society to climate change (such as coastal protection and adaption of agricultural practices to more arid and hot conditions) will be discussed.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes</td></tr>
</table>

Students acquire transferable and key skills in this module.

By the end of this module, students should be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- advance a knowledge-based opinion on the complex module topics, including: impact of climate change on the natural environment over geological timescales and since the industrial revolution, and the policy framework in which environmental decisions are made internationally;
- work effectively in a team environment and undertake data interpretation;
- discuss approaches to minimize habitat destruction.

Indicative Literature

The course is based on a self-contained, detailed set of online lecture notes.

Ruddiman, William F. Earth's Climate (2001). Past and future. New York: Macmillan.

Usability and Relationship to other Modules

- This module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Written examination	Duration: 60 min.
Scope: All intended learning outcomes of the module	Weight: 100%

7.26.2.7 Extreme Natural Hazards, Disaster Risks, and Societal Impact

<table>
<tr><td colspan="2">Module Name
Extreme Natural Hazards, Disaster Risks, and Societal Impact</td><td>Module Code
JTBQ-BQ-008</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components
Number | Name | Type | CP
JTBQ-008 | Extreme Natural Hazards: Disaster Risks, and Societal Impact | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator

L. Thomsen</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (Fall)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Lecture (17.5 hours)
• Private study (45 hours)

Workload

62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage of the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizons with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
Extreme natural events increasingly dominate global headlines, and understanding their causes, risks, and impacts, as well as the costs of their mitigation, is essential to managing hazard risk and saving lives. This module presents a unique, interdisciplinary approach to disaster risk research, combining natural science and social science methodologies. It presents the risks of global hazards and natural disasters such as volcanoes, earthquakes, landslides, hurricanes, precipitation floods, and space weather, and provides real-world hazard and disaster case studies from Latin America, the Caribbean, Africa, the Middle East, Asia, and the Pacific.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, student should be able to

• use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;</td></tr>
</table>

- advance a knowledge-based opinion on the complex module topics, including how natural processes affect and interact with our civilization, especially those that create hazards and disasters;
- distinguish the methods scientists use to predict and assess the risk of natural disasters;
- discuss the social implications and policy framework in which decisions are made to manage natural disasters;
- work effectively in a team environment.

Indicative Literature

The course is based on a self-contained, detailed set of online lecture notes.

Ismail-Zadeh, Alik, et al., eds (2014). Extreme natural hazards, disaster risks and societal implications. In *Special Publications of the International Union of Geodesy and Geophysics Vol. 1.* Cambridge: Cambridge University Press.

Usability and Relationship to other Modules

- The module is a mandatory elective module of the Big Questions area, that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules)
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute such knowledge and competences to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Written examination — Duration: 60 min.
Scope: All intended learning outcomes of the module — Weight: 100%

7.26.2.8 International Development Policy

<table>
<tr><td colspan="2">Module Name
International Development Policy</td><td>Module Code
JTBQ-BQ-009</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components
Number | Name | Type | CP
JTBQ-009 | International Development Policy | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator

C. Knoop</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs, except IEM</td><td colspan="2">Mandatory Status

• Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking, and a proficient handling of data sources</td><td>Frequency

Annually (Fall)</td><td>Forms of Learning and Teaching

• Lecture (17.5 hours)
• Presentations
• Private study (45 hours)</td></tr>
<tr><td>Duration

1 semester</td><td>Workload

62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Critically following media coverage of the module's topics in question.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal, and environmental contexts of the global issues and challenges of the coming decades. BQ modules intend to raise awareness of those challenges and broaden students' horizon with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become informed and responsible citizens in a global society.
We live in a world where still a large number of people still live in absolute poverty without access to basic needs and services, such as food, sanitation, health care, security, and proper education. This module provides an introduction to the basic elements of international development policy, with a focus on the relevant EU policies in this field and on the Sustainable Development Goals/SDGs of the United Nations. The students will not only learn about the tools applied in modern development policies, but also about the critical aspects of monitoring and evaluating the results of development policy. Module-related oral presentations and debates will enhance the students' learning experience.</td></tr>
</table>

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, the student should be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- breakdown the complexity of modern development policy;
- identify, explain, and evaluate the tools applied in development policy;
- formulate well-justified criticism of development policy;
- summarize and present a module-related topic in an appropriate verbal and visual form.

Indicative Literature

Francis Fukuyama (2006). The end of history and the last man. New York: Free Press.

Kingsbury, McKay, Hunt (2008). International Development.Issues and challenges. London: Palgrave.

A.Sumner, M.Tiwari (2009) After 2015: International Development Policy at a crossroad. New York: Palgrave Macmillan.

Graduate Institute of International Development, G. Carbonnier eds. (2001). International Development Policy: Energy and Development. New York:Palgrave Macmillan.

John Donald McNeil. International Development: Challenges and Controversy. Sentia Publishing,e-book.

Usability and Relationship to other Modules

- This module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Presentation
Duration: 10 minutes per student
Scope: All intended learning outcomes of the module
Weight: 100%

7.26.2.9 Sustainable Value Creation with Biotechnology. From Science to Business

<table>
<tr><td colspan="2">Module Name
Sustainable Value Creation with Biotechnology. From Science to Business.</td><td>Module Code
JTBQ-BQ-011</td><td>Level (type)
Year 3
(Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components</td></tr>
<tr><td>Number</td><td>Name</td><td colspan="2">Type</td><td>CP</td></tr>
<tr><td>JTBQ-011</td><td>Sustainable Value Creation with Biotechnology. From Science to Business</td><td colspan="2">Lecture - Tutorial</td><td>2.5</td></tr>
<tr><td>Module Coordinator
Marcelo Fernandez Lahore</td><td colspan="2">Program Affiliation
• Jacobs Track - Big Questions</td><td colspan="2">Mandatory Status
• Mandatory elective for students of all undergraduate study except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements
Pre-requisites
☒ None
Co-requisites
☒ None
Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues on bio-based value creation
• media literacy, critical thinking and a proficient handling of data sources</td><td>Frequency
Annually (Spring)</td><td colspan="2">Forms of Learning and Teaching
• Lecture and Tutorial (17.5 hours)
• Private study (45 hours)</td></tr>
<tr><td>Duration
1 semester</td><td colspan="2">Workload
62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation
https://www.ctsi.ucla.edu/researcher-resources/files/view/docs/EGBS4_Kolchinsky.pdf
https://link.springer.com/article/10.1057/jcb.2008.27
https://sustainabledevelopment.un.org/content/documents/21252030%20Agenda%20for%20Sustainable%20Development%20web.pdf</td></tr>
</table>

Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal and environmental contexts of the global issues and challenges of the coming decades. The BQ modules intend to raise awareness of those challenges and broaden the students' horizon with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules support students in their development to become an informed and responsible citizen in a global society.
This module has a particular focus on the role that Biotechnology and Biorefining is expected to play in social, economic and environmental contexts.

To deliver such a vision the module will prepare students to extract value form Biotechnology and associated activities. This will be done in the form of business cases that will be systematically developed by students alongside the development of the module. In this way, students will develop entrepreneurial skills while understanding basic business-related activities that are not always present in a technical curriculum. Case development will also provide students with the possibility of understanding the social, economic, environmental impact that Biotechnology and Biorefining can deliver in a Bio-Based Economy. The knowledge and skills gained through this module are in direct and indirect support of the UN 2030 Agenda for Sustainable Development: "Transforming our World".

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, the students should be able to

- design and develop a Business Case based on the tools provided by modern Biotechnology;
- explain the interplay between Science, Technology and Economics / Finance;
- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- work effectively in a team environment and undertake data interpretation and analysis;
- discuss approaches to value creation in the context of Biotechnology and Sustainable Development;
- explain the ethical implications of technological advance and implementation;
- demonstrate presentation skills.

Indicative Literature

Springham, D., V. Moses & R.E. Cape (1999). Biotechnology – The Science and the Business. 2nd. Ed. Boca Raton: CRC Press.

Kornberg, Arthur (2002). The Golden Helix: Inside Biotech Ventures. Sausalito, CA: University Science Books.

UNESCO, Director-General. (2017). UNESCO moving forward the 2030 Agenda for Sustainable Development. Retrieved from https://unesdoc.unesco.org/ark:/48223/pf0000247785

Usability and Relationship to other Modules

- The module is a mandatory elective module in the Big Questions area, which is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute their knowledge and competencies to class discussions and activities.

Examination Type: Module Examination

Assessment Component 1: Term Paper	Length:1.500 – 3.000 words Weight: 75%
Scope: Intended learning outcomes of the module (1-6)	
Assessment Component 2: Presentation	Duration: 10-15 min.

Weight: 25%

Scope: Intended learning outcomes of the module (2-7)

7.26.2.10 Gender and Multiculturalism. Debates and Trends in Contemporary Societies

<table>
<tr><td colspan="3">Module Name
Gender and Multiculturalism. Debates and Trends in Contemporary Societies</td><td>Module Code
JTBQ-BQ-013</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
5.0</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
JTBQ-013 | Gender and Multiculturalism: Debates and Trends in Contemporary Societies | Lecture | 5.0</td></tr>
<tr><td>Module Coordinator

J. Price</td><td colspan="3">Program Affiliation

• Big Questions Area: All undergraduate study programs</td><td colspan="2">Mandatory Status

Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• The ability and openness to engage in interdisciplinary issues of global relevance
• Media literacy, critical thinking and a proficient handling of data sources</td><td>Frequency

Annually (Fall)</td><td colspan="2">Forms of Learning and Teaching

• Lectures (17.5 hours)
• Project work (90 hours)
• Private study (17.5 hours)</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Critical following of the media coverage on the module's topics in question.</td></tr>
<tr><td colspan="6">Content and Educational Aims

All "Big Questions" (BQ) modules deal with the economic, technological, societal and environmental contexts of the global issues and challenges of the coming decades. The BQ modules intend to raise awareness of those challenges and broaden the students' horizon with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules are relevant for every university graduate in order to become an informed and responsible citizen in a global society.
The objective of this module is to introduce and familiarize students with the current debates, trends and analytical frameworks pertaining how gender is socially constructed in different cultural zones. Through lectures, group discussions and reflecting upon cultural cases, students will familiarize themselves with the current trends and the different sides of ongoing cultural and political debates that shape cultural practices, policies and discourses. The module will zoom-in on topics such as: cultural identity; the social construction of gender; gender fluidity and its backlash; gender and human rights; multiculturalism as a perceived threat in plural societies, among others. Students will be provided with opportunities for reflection and to ultimately develop informed opinions concerning topics that are continue to define some of the most contested cultural debates of contemporary societies. Furthermore, participants will engage their ideas in "hands on" projects aimed at moving</td></tr>
</table>

the needle from mere reflection by conducting "action-research" that will inform the outcomes of their course projects.

Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

- use their disciplinary factual and methodological knowledge to reflect on interdisciplinary questions by comparing approaches from various disciplines;
- summarize and evaluate the current cultural, political and legal debates concerning the social construction of gender in contemporary societies;
- reflect and develop informed opinions concerning the current debates and trends that are shaping ideas of whether multiculturalism ideals are realistic in pluralist western societies, or whether multiculturalism is a failed project;
- identify, explain and evaluate the role that societal forces, such as religion, socio-economic, political and migratory factors play in the construction of gendered structures in contemporary societies;
- develop a well-informed perspective concerning the interplay of science and culture in the debates around gender fluidity;
- deconstruct and reflect on the intersectionality between populist/nationalist discourses and gender discrimination;
- reflect and propose societal strategies and initiatives that attempt to answer the big questions presented in this module regarding gendered and cross-culturally-based inequalities;
- complete a self-designed project, collect and distill information from an "action-research" perspective;summarizing the process in a suitable reporting format;
- consider the application of an algorithm for group formation (not mandatory);
- overcome general teamwork problems in order to perform well-organized project work.

Indicative Literature

Biological Limits of Gender Construction Author(s): J. Richard Udry

Source: American Sociological Review , Jun., 2000, Vol. 65, No. 3 (Jun., 2000), pp. 443- 457. Published by: American Sociological Association Stable URL: https://www.jstor.org/stable/2657466

The Development of Gendered Interests and Personality Qualities From Middle Childhood Through Adolescence: A Biosocial Analysis. Susan M. McHale, Aryn M. Dotterer, Ji-Yeon Kim, Ann C. Crouter and Alan Booth. Child Development, March/April 2009, Volume 80, Number 2, Pages 482–495

Factors influencing attitudes to violence against women. Michael Flood and Bob Pease. Trauma, Violence, & ABuse, Vol. 10, No. 2, April 2009 125-142 dOi: 10.1177/1524838009334131. 2009 sAge Publications

Gender and Anti-immigrant Attitudes in Europe. Aaron Ponce (2017) Socius: Sociological Research for a Dynamic World. Volume 3: 1–17. Reprints and permissions: sagepub.com/journalsPermissions.nav

Usability and Relationship to other Modules

- The module is a mandatory elective module of the Big Questions area, that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules)
- Students are encouraged to relate the content of their previous modules to the topics of this module and contribute such knowledge and competences to class discussions and activities.

Examination Type: Module Examination

Assessment Type: Team Project

Weight: 100%

Scope: All intended learning outcomes of the module

7.26.2.11 The Challenge of Sustainable Energy

<table>
<tr><td colspan="2">Module Name
The Challenge of Sustainable Energy</td><td>Module Code
JTBQ-BQ-014</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components

Number | Type | CP
JTBQ-014 | The Challenge of Sustainable Energy | Lecture | 2.5</td></tr>
<tr><td>Module Coordinator

K. Smith Stegen</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs</td><td colspan="2">Mandatory Status

Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="3">Entry Requirements

Pre-requisites | Co-requisites | Knowledge, Abilities, or Skills
☒ None | ☒ None | • Ability to read texts from a variety of disciplines</td><td>Frequency

Annually (Spring)

Duration

1 semester</td><td>Forms of Learning and Teaching

• Lectures and Group Exercises

Workload

62.5 hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Reflect on their own behavior and habits with regard to sustainability.</td></tr>
<tr><td colspan="5">Content and Educational Aims

All “Big Questions” (BQ) modules deal with the economic, technological, societal and environmental contexts of the global issues and challenges of the coming decades. The BQ modules intend to raise awareness of those challenges and broaden the students’ horizon with applied problem solving beyond the borders of their own disciplines. Knowledge and skills offered in the interdisciplinary BQ modules are relevant for every university graduate in order to become an informed and responsible citizen in a global society.
How can wide-scale social, economic and political change be achieved? This module examines this question in the context of encouraging “sustainability”. To address global warming and environmental degradation, humans must adopt more sustainable lifestyles. Arguably, the most important change is the transition from conventional fuels to renewable sources of energy, particularly at the local, country and regional levels. The main challenge to achieving an “energy transition” stems from human behavior and not from a lack of technology or scientific expertise. This module thus examines energy transitions from the perspective of the social sciences, including political science, sociology, psychology, economics and management. To understand the drivers of and obstacles to technology transitions, students will learn the “Multi-Level Perspective”. Some of the key questions explored in this module include: What is meant by sustainability? Are renewable energies “sustainable”? How can a transition to renewable energies be encouraged? What are the main social, economic, and political challenges? How can these (potentially) be overcome? The aim of the course is to provide students with the tools for reflecting on energy transitions from multiple perspectives.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

Students acquire transferable and key skills in this module.

By the end of this module, students will be able to

• articulate the history of the sustainability movement and the major debates;</td></tr>
</table>

- identify different types of renewable energies;
- explain the multi-level perspective (MLP), which models technology innovations and transitions;
- summarize the obstacles to energy transitions;
- compare a variety of policy mechanisms for encouraging renewable energies.

Usability and Relationship to other Modules

- The module is a mandatory elective module of the Big Questions area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).
- For students interested in sustainability issues, this module complements a variety of modules from different programs, such as "International Resource Politics" (IRPH/SMP), "Environmental Science" (EES), "General Earth and Environmental Sciences" (EES), and "Renewable Energies" (Physics).

Examination Type: Module Examination

Assessment Type: Written Examination

Duration: 60 min
Weight: 100%

Scope: All intended learning outcomes of the module

7.26.2.12 State, Religion and Secularism

<table>
<tr><td colspan="2">Module Name
State, Religion and Secularism</td><td>Module Code
JTBQ-BQ-015</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
2.5</td></tr>
<tr><td colspan="5">Module Components

Number | Type | CP</td></tr>
<tr><td>JTBQ-015</td><td colspan="2">State, religion and secularism</td><td>Lecture</td><td>2.5</td></tr>
<tr><td>Module Coordinator

Manfred O. Hinz</td><td colspan="2">Program Affiliation

• Big Questions Area: All undergraduate study programs</td><td colspan="2">Mandatory Status

Mandatory elective for students of all undergraduate study programs, except IEM</td></tr>
<tr><td colspan="2" rowspan="2">Entry Requirements

Pre-requisites
☒ None

Co-requisites
☒ None

Knowledge, Abilities, or Skills
• Ability to read texts from a variety of disciplines</td><td>Frequency

Annually (Spring)</td><td colspan="2">Forms of Learning and Teaching

• Lectures and Group Exercises</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

62.5 Hours</td></tr>
<tr><td colspan="5">Recommendations for Preparation

Reflect on the situation and role in respective home-country</td></tr>
<tr><td colspan="5">Content and Educational Aims

The relationship between state and religion has been a matter of concern in most if not all societies. Is religion above the state, or is it to the state to determine the place of religion? What does secularity mean? To what extent will religion accept secularity? Where does the idea of secularity come from? The course State, religion, secularism will search for answers to questions of this nature. After introducing to the topic and looking at some legal attempts to regulate the relationship between state and religion, the focus will be, on the one hand, on Christianity and secularity and, on Islam and secularity, on the other. Depending on the interest of participants, other religions and their relationships to states of relevance can be added.</td></tr>
<tr><td colspan="5">Intended Learning Outcomes

By the end of this course, students should be able
• To understand the basic problems that have led to different models to regulate the relationship between the state and religion;
• To reflect critically the situation of state and religion in selected countries;
• To assess the values behind the concept of democracy and human rights;
• To use the acquired knowledge to strengthen the capacity towards respect for others and tolerance.</td></tr>
<tr><td colspan="5">Usability and Relationship to other Modules

• The module is a mandatory elective module of the Big Questions area that is part of the Jacobs Track (Methods and Skills modules; Community Impact Project module; Language modules; Big Questions modules).</td></tr>
</table>

- For students interested in State, Religion and secularism, this module complements modules from other programmes, such as IRPH and SMP

Examination Type: Module Examination

Assessment Type: Term paper — Length:1.500 – 3.000 words
Weight: 100%

Scope: All intended learning outcomes of the module.

7.26.3 Community Impact Project

<table>
<tr><td colspan="3">Module Name
Community Impact Project</td><td>Module Code
JTCI-CI-950</td><td>Level (type)
Year 3 (Jacobs Track)</td><td>CP
5</td></tr>
<tr><td colspan="6">Module Components

Number | Name | Type | CP
JTCI-950 | Community Impact Project | Project | 5</td></tr>
<tr><td colspan="2">Module Coordinator

CIP Faculty Coordinator</td><td colspan="2">Program Affiliation

• All undergraduate study programs except IEM</td><td colspan="2">Mandatory Status

Mandatory for all undergraduate study programs except IEM</td></tr>
<tr><td colspan="3" rowspan="2">Entry Requirements

Pre-requisites

☒ at least 15 CP from CORE modules in the major

Co-requisites

☒ None

Knowledge, Abilities, or Skills

• Basic knowledge of the main concepts and methodological instruments of the respective disciplines</td><td>Frequency

Annually (Fall)</td><td colspan="2">Forms of Learning and Teaching

• Introductory, accompanying, and final events: 10 hours
• Self-organized teamwork and/or practical work in the community: 115 hours</td></tr>
<tr><td>Duration

1 semester</td><td colspan="2">Workload

125 hours</td></tr>
<tr><td colspan="6">Recommendations for Preparation

Develop or join a community impact project before the 5th semester based on the introductory events during the 4th semester by using the database of projects, communicating with fellow students and faculty, and finding potential companies, organizations, or communities to target.</td></tr>
<tr><td colspan="6">Content and Educational Aims

CIPs are self-organized, major-related, and problem-centered applications of students' acquired knowledge and skills. These activities will ideally be connected to their majors so that they will challenge the students' sense of practical relevance and social responsibility within the field of their studies. Projects will tackle real issues in their direct and/or broader social environment. These projects ideally connect the campus community to other communities, companies, or organizations in a mutually beneficial way.
Students are encouraged to create their own projects and find partners (e.g., companies, schools, NGOs), but will get help from the CIP faculty coordinator team and faculty mentors to do so. They can join and collaborate in interdisciplinary groups that attack a given issue from different disciplinary perspectives.
Student activities are self-organized but can draw on the support and guidance of both faculty and the CIP faculty coordinator team.</td></tr>
<tr><td colspan="6">Intended Learning Outcomes

The Community Impact Project is designed to convey the required personal and social competencies for enabling students to finish their studies at Jacobs as socially conscious and responsible graduates (part of the Jacobs mission) and to convey social and personal abilities to the students, including a practical awareness of the societal context and relevance of their academic discipline.

By the end of this project, students should be able to</td></tr>
</table>

- understand the real-life issues of communities, organizations, and industries and relate them to concepts in their own discipline;
- enhance problem-solving skills and develop critical faculty, create solutions to problems, and communicate these solutions appropriately to their audience;
- apply media and communication skills in diverse and non-peer social contexts;
- develop an awareness of the societal relevance of their own scientific actions and a sense of social responsibility for their social surroundings;
- reflect on their own behavior critically in relation to social expectations and consequences;
- work in a team and deal with diversity, develop cooperation and conflict skills, and strengthen their empathy and tolerance for ambiguity.

Indicative Literature

Not specified

Usability and Relationship to other Modules

- Students who have accomplished their CIP (6th semester) are encouraged to support their fellow students during the development phase of the next year's projects (4th semester).

Examination Type: Module Examination

Project, not numerically graded (pass/fail)
Scope: All intended learning outcomes of the module

7.26.4 Language Modules

The descriptions of the language modules are provided in a separate document, the “Language Module Handbook” that can be accessed from here: https://www.jacobs-university.de/study/learning-languages

8 Appendix

8.1 Intended Learning Outcomes Assessment-Matrix

Computer Science (BSc.)					Introduction to Computer Science	Introduction to Robotics and Intelligent Systems	Programming in C and C++	Algorithms and Data Structures	Databases and Web Services	Software Engineering	Operating Systems	Automata, Computability, and Complexity	Computer Networks	Legal and Ethical Aspects of Computer Science	Secure and Dependable Systems	Academic Skills in Computer Science	Computer Graphics	Image Processing	Distributed Algorithms	Web Application Development	Bachelor Thesis	Calculus and Elements of Linear Algebra 1	Calculus and Elements of Linear Algebra 2	Probability and Random Processes	Discrete Mathematics / Numerical Methods	Internship	JT Big Questions	JT Community Impact	JT Language
Semester					1	2	1	2	3	4	3	4	3	3	4	4	5	6	5	6	6	1	2	3	4	5	5/6	5	1-4
Mandatory/mandatory elective					m	m	m	m	m	m	m	m	m	m	m	me	me	me	me	me	m	m	m	m	me	m	me	m	me
Credits					7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5	5	2.5	5	2.5	5	5	5	5	15	5	5	5	5	15	10	5	10
	Competencies*																												
Program Learning Outcomes	**A**	**E**	**P**	**S**																									
Work professionally in the highly dynamic computer science field and enter graduate programs related to computer science.	x	x	x		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Apply fundamental concepts of computer science while solving problems.	x	x			x	x	x	x	x	x	x	x	x		x		x	x	x	x	x					x		x	
Think in an analytic way at multiple levels of abstraction.	x	x	x	x	x	x	x	x	x	x	x	x	x		x	x	x	x	x	x	x	x	x	x	x				
Develop, analyze and implement algorithms using modern software engineering methods.	x	x			x		x	x	x	x					x		x	x		x	x								
Understand the characteristics of a range of computing platforms and their advantages and limitations.	x	x			x	x					x		x				x	x			x								
Choose from multiple programming paradigms, languages and algorithms in order to solve a given problem in an adequate way.	x	x			x	x	x	x	x	x	x		x		x		x	x	x	x	x								
Describe the fundamental theory of computation and computability.	x				x			x				x																	
Apply the necessary mathematical methods.	x																					x	x	x	x				
Recognize the context in which computer systems operate, including interactions with people and the physical world.	x	x	x	x	x	x					x			x	x	x	x	x	x	x	x					x		x	
Describe the state of published knowledge in their field or a specialization within it.	x	x	x						x	x	x	x	x		x	x	x	x	x	x	x								
Analyze and model real life scenarios in organizations and industries using contemporary techniques of computer science, also taking methods and insights of other disciplines into account.	x	x	x		x	x			x	x	x		x		x		x	x	x	x	x					x		x	
Communicate solutions of problems in computer science in both spoken and written form appropriately to specialists and non-specialists.	x	x	x	x						x						x					x					x		x	
Draw scientifically-founded conclusions that consider social, professional, scientific and ethical aspects.	x	x	x	x					x	x	x		x	x	x	x	x	x	x	x	x					x	x	x	
Work effectively in a diverse team and to take responsibility in a team.	x	x	x	x						x																x	x	x	x
Take responsibility for their own learnig, personal and professional development and role in society, reflecting on their practice and evaluating critical feedback.														x		x					x					x	x	x	
Adhere to and defend ethical, scientific and professional standards.	x	x	x	x						x				x	x	x										x		x	
Assessment Type																													
Oral examination																													
Written examination					x	x	x	x	x	x	x	x	x		x		x	x	x	x		x	x	x	x				
Project										x						x			x									x	
Term papaer, essay																													
(Lab) Report																										x			
Poster presentation														x															
Presentation																					x								
Thesis																					x								
Various																											x		x
Module achievements/bonus achievements					x	x	x	x			x																		x

*Competencies: A-scientific/academic proficiency; E-competence for qualified employment; P-development of personality; S-competence for engagement in society

Figure 4: Intended Learning Outcomes Assessment-Matrix