Stage I Studies

Compulsory Subjects

Mathematics

Course: Mathematics
Lecturer/Instructor: Prof. K. Engel, Prof. D. Lau
Term: Four-semester-Contents, start every winter semester
Hours per Week: 6+2, 6+2, 4+2, 4+2
Credits: 12, 12, 8, 8
Preconditions: None
Elaboration/Examination: Oral examination
Language: German
Goals: Introduction into higher mathematics
Description: Basic notions in mathematics; Systems of linear equations, matrices and determinants; Vector and affine spaces; Eigenvalues and Eigenvectors; Curves and surfaces of second order; Linear programming; Sequences and series; Functions; Differential calculus; Integral calculus; Calculus of several variables; Series of functions; Numerical mathematics; Differential equations; Combinatorics; Probability theory; Statistics
Recommended Readings: Announced at the beginning of the lecture

Practical Computer Science

Course: Algorithms and Data Structures
Lecturer/Instructor: Prof. Forbrig, Prof. Hantzschmann, Prof. Riedewald (alternating)
Term: Part I: every Winter semester
Part II: every Summer semester
Hours per Week: 3 (lectures); 3 (exercises); small individual project
Credits: 8+8
Preconditions: None
Elaboration/Examination: Evaluated project and exercises; part of the exam in Practical Computer Science
Language: German
Goals: Introduction to programming in small, programming exercises
Description: Basic programming techniques; Basic algorithmic constructs; Test and verification; Basic data structures as Abstract Data Types - their definition, implementation and application; Informal introduction to the Theory of Algorithms; Special algorithms; Sorting and searching
Recommended Readings: Announced at the beginning of the lecture

Course: Software Engineering
Lecturer/Instructor: Prof. Forbrig
Term: Every Winter semester
Hours per Week: 2 (lectures); 1 (exercise); additional project work till summer
Credits: 6
Preconditions: Algorithms and Data Structures
Elaboration/Examination: Individual exercises; project work
Language: German
Goals: Introduction to software engineering techniques, methods and tools with emphasis on an object-oriented approach. Software development in teams.
Description: Software life-cycle models; Principles of software engineering; Specification techniques; Prototyping; Structured analysis and design; Object-oriented analysis and design; Case-tools; Development of software projects
Course: Operating Systems  
Lecturer/Instruction: Dr. Hochberger  
Term: Every Summer semester  
Hours per Week: 2 (lectures); 2 (exercises)  
Credits: 4  
Preconditions: CS I+II and Processor Architectures  
Elaboration/Examination: Oral examination  
Language: German  
Goals: Understanding of Operating Systems and their mechanisms  
Description: Conceptions and structures of operating systems: Introduction (targets, functions, Terms, definition, components, history); File systems; Processes; Memory management; I/O; Security; Process communication/deadlock; Case examples etc.  

Theoretical Computer Science  
Course: Theoretical Foundations of Computer Science  
Lecturer/Instruction: Prof. Hantzschiann, Prof. Brandstädt (alternating)  
Term: Part I: every Summer semester  
Part II: every Winter semester  
Hours per Week: 2 (lectures); 1 (exercise)  
Credits: 4+4  
Preconditions: None  
Elaboration/Examination: Final examination (oral) after 4th semester  
Language: German  
Goals: Introduction to basic techniques in computability, efficiency of computation, formal Languages and finite automata  
Description: Machine models; Turing machines and computable functions; Decidability and enumerability; Undecidable problems; Polynomial time computability; P and NP; Concrete problems; Efficiency of computations; Context free Languages; Chomsky-hierarchy; Finite automata and regular Languages  

Course: Theory of Programming Languages  
Lecturer/Instruction: Prof. Riedewald  
Term: Every Summer semester  
Hours per Week: 2 (lectures); 1 (exercise)  
Credits: 4  
Preconditions: Knowledge in the Theory of Formal Languages, programming experience in a higher programming Language  
Elaboration/Examination: Part of the exam in Theoretical Computer Science  
Language: German  
Goals: Introduction to Formal Semantics, complete formal description of programming Languages  
Description: Concrete and abstract syntax; Introduction to Formal Semantics; Complete formal description of programming Languages (Attribute Grammars,...)  
Recommended Readings: Announced at the beginning of the lecture.
Technical Computer Science

Course: Physically - electronic Bases of Computer Science
Lecturer/Instructor: Dr. Axel Rennau
Term: Part I: every Winter semester
        Part II: every Summer semester
Hours per Week: 2 (lectures); 1 (exercise)
Credits: 4+4
Preconditions: None
Elaboration/Examination: Written examination
Language: German
Goals: Understanding of physical fundamentals of electrics and semiconductors
Description: Introduction to electronic basic relations as well as characteristics of digital circuits, divided into the following parts:
1. Bases of electric engineering (electric circuits, basic relations, field quantities, passive and active one ports, switching behaviour, network calculations)
2. Semiconductor electronics (materials, energy-band model, p-n junction, crystal diodes, bipolar transistors, field-effect transistors, circuiting applications)
3. Integrated circuits, digital circuit technologies (basic circuits, combinatorial and sequential systems, digital memory, signal generation and data transformation, user specified circuits) display technology, opto-electronics, operational amplifier
4. If necessary: Alternating current circuits (Symbolic method of electric engineering - complex operational calculus, alternating current performance, transient characteristic, amplitude and phase response, critical frequencies, fourier-analysis)

Recommended Readings:
Paul, Reinhold: Elektrotechnik und Elektronik für Informatiker
Band 1 Grundgebiete der Elektrotechnik
Band 2 Grundgebiete der Elektronik, B.G. Teubner Stuttgart;
Kühn, Eberhard: Handbuch TTL - und CMOS – Schaltungen, Hüthig Verlag
Heidelberg;
Tietze, U.; Schenk, Ch.: Halbleiter – Schaltungstechnik, Springer-Verlag.

Course: Computer Systems I
Lecturer/Instructor: D. Tavangarian / C. Cap (alternating)
Term: Every Winter semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 5
Preconditions: None
Elaboration/Examination: Oral examination
Language: German
Goals: Understanding the basics of digital systems
Description: Basics of the functional structures and the design of digital circuits:
Boolean algebra; Boolean functions; Combinatorial circuits (minimization and implementation procedure); Coding procedure: digital number notation and processing in computers; Memory elements and techniques (flip-flop, RAMs, ROM, EPROMs, EEPROMs etc.); Switching devices: concepts, design procedure, optimization algorithms; Implementation possibilities of components in digital computers

Recommended Readings:
H. Liebig, Th. Flik: Rechnerorganisation - Prinzip, Struktur, Algorithmen; Springer-Verlag (2. Auflage 1993);
S. Hentschke: Grundzüge der Digitaltechnik, B. G. Teubner Stuttgart (1998);
**Course:** Computer Systems II  
**Lecturer/Instruction:** D. Tavangarian / C. Cap (alternating)  
**Term:** Every Summer semester  
**Hours per Week:** 2 (lectures); 2 (exercises)  
**Credits:** 5  
**Preconditions:** Computer System I  
**Elaboration/Examination:** Oral examination  
**Language:** German  
**Goals:** Understanding the basic structures of microprocessors  
**Description:** Introduction to the essential structure of a microprocessor system, its basic components (like control units, arithmetic/logic units, bus structures, I/O-units and instruction formats); Architectural aspects and the programming of a microprocessor (with the help of a hypothetical microprocessor); Different microprocessor conceptions  
**Recommended Readings:**  
H. Liebig, Th. Flik: Rechnerorganisation - Prinzip, Struktur, Algorithmen; Springer-Verlag (2. Auflage 1993);  
S. Hentschke: Grundzüge der Digitaltechnik, B. G. Teubner Stuttgart (1998);  
N. Wirth: Digital Circuit Design An Introductory Textbook, Springer-Verlag (1995);  
W. Schaffmann, R. Schmitz: Technische Informatik, Band 2 Grundlagen der Computertechnik, Springer-Verlag-Lehrbuch (1992);  

---

**Course:** Processor Architectures  
**Lecturer/Instruction:** D. Tavangarian  
**Term:** Every Winter semester  
**Hours per Week:** 2 (lectures); 2 (exercises)  
**Credits:** 4  
**Preconditions:** Computer Systems I+II  
**Elaboration/Examination:** Oral examination  
**Language:** German  
**Goals:** Processor architectures and organization  
**Description:** Basic structure of a microprocessor, its advanced components and their functions; Selection of additional material derives from modern microprocessors (CISC as well as RISC types) and their system blocks; Pipelining, storage and cache management, memory protection, bus architectures, principles of the input and output, co-processor systems and more; Case studies about the characteristics of these and analysis of the newest systems  
**Recommended Readings:**  
W. Oberschelp, G. Vossen: Rechneraufbau und Rechnerstrukturen, R. Oldenbourg Verlag (6. Auflage 1994);  
W. Erhardt: Rechnerarchitektur, Einführung und Grundlagen, B. G. Teubner Stuttgart (1995);  
W. Schaffmann, R. Schmitz: Technische Informatik, Band 2 Grundlagen der Computertechnik, Springer-Verlag-Lehrbuch (1992);  
H. Bähring, Mikrorechnersysteme, Springer-Verlag (1991);  

---

**Introduction to scientific learning and studying techniques**  
**Course:** Introduction to scientific learning and studying techniques.  
**Lecturer:** Prof. Cap  
**Term:** Every Term  
**Hours per Week:** Block of 14 hours at the beginning or at the end of the Term  
**Credits:** 1  
**Preconditions:** None  
**Elaboration/Examination:** Practical exercises  
**Language:** German  
**Goals:** To get accustomed to the fundamental techniques of scientific work  
**Description:**  
* What is science?  
* Behaviour in the scientific world  
* How to deliver a scientific talk  
* How to write a scientific paper  
**Recommended Readings:** None
**Stage II Studies**

**Compulsory Subjects**

<table>
<thead>
<tr>
<th>Course</th>
<th>Computer Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Dr. rer. nat. habil. S. Adomssent</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Vordiplom</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Final examination after the 5th semester</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Introduction to basic Techniques of the Computer Networks, TCP/IP-protocols</td>
</tr>
<tr>
<td>Description</td>
<td>Exchanging Information; OSI Model; Network Media (Cable Media, Wireless Media); Signal Transmission; Network Topology; Real Networks (Ethernet, Token Ring, FDDI, DQDB); Network Protocols; Network Connectivity (Bridges, Router, Gateways); Internet Protocols (TCP/IP, ICMP, ARP, RIPv); Managing and Troubleshooting of Networks</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Computer Graphics I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Jackèl, Prof. Schumann (alternating)</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>2 (lectures); 2 (exercises)</td>
</tr>
<tr>
<td>Credits</td>
<td>4</td>
</tr>
<tr>
<td>Preconditions</td>
<td>None</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Verbal examination</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Introduction into the fundamentals and principles of 3D computer graphics</td>
</tr>
<tr>
<td>Description</td>
<td>Graphics standards (GKS, PHIGS, OpenGL); 2D/3D-transformations; Rendering methods; Computer graphics hardware; Color systems and perception of color; Geometric modeling; User interfaces; etc.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Fundamentals of Databases I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Andreas Heuer</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>3 (lectures); 1 (exercise)</td>
</tr>
<tr>
<td>Credits</td>
<td>6</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Algorithms and Data Structures</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Individual exercises, final examination</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Understanding the principles of database systems; Introduction to relational database theory, schema design, query Languages</td>
</tr>
<tr>
<td>Description</td>
<td>Basic concepts of database systems, architectures and database models; Database schema definition and design; Relational database schema design, query and update operations; Relational database Languages; Database application programming basics; Views; Data security and integrity</td>
</tr>
<tr>
<td>Course:</td>
<td>Modeling and Simulation I: Basic Methods of Modeling and Simulation</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Uhrmacher</td>
</tr>
<tr>
<td>Term:</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>3 (lectures); 1 (exercises)</td>
</tr>
<tr>
<td>Credits:</td>
<td>6</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Vordiplom</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Programming exercises; Oral examination</td>
</tr>
<tr>
<td>Language:</td>
<td>German or English /Slides partly in English</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introduction to basic concepts for the simulation with an emphasis on discrete systems</td>
</tr>
<tr>
<td>Description:</td>
<td>Unified Modeling Language; Hierarchical, Modular Model Construction; Queueing Systems; Flow Shop Simulation with Arena; Petri Nets; Qualitative Simulation; Inductive Methods; Basics of Continuous Simulation; Basics of Parallel Simulation; Basics of Agent-Oriented Simulation</td>
</tr>
</tbody>
</table>
Course: Compiler Techniques
Lecturer/Instruction: Prof. Riedwald
Term: Every Summer semester
Hours per Week: 3 (lectures); 1 (exercise)
Credits: 6
Preconditions: Knowledge in the Theory of Formal Languages and the Theory of Automata, programming experience in a higher programming Language and in an assembler Language, introductory knowledge in Software Engineering
Elaboration/Examination: Exam
Language: German
Goals: Introduction to compiler architecture and to selected tools of compiler generation
Description: Compilers as an example of big software systems; Architecture of compilers; Introduction to automatic compiler generation
Recommended Readings: Announced at the beginning of the lecture
--------------------------------------------------------------------------------------------------
Stage II Subjects

Optional Compulsory Subjects

Practical Computer Science

Programming Languages and Compiler Construction

Course: Programming Language ADA  
Lecturer/Instructor: Prof. Riedewald  
Term: Every Winter semester  
Hours per Week: 1 week at the end of the semester  
Credits: 3  
Preconditions: Programming experiences in a higher programming Language  
Elaboration/Examination: Individual exercises  
Language: German  
Goals: Introduction to ADA  
Description: Lectures on architecture and main Language constructs enabling imperative and object-oriented programming; Computer exercises  
Recommended Readings: Announced at the beginning of the lecture

Course: Attribute Grammars and their Applications  
Lecturer/Instructor: Prof. Riedewald  
Term: Winter semester; irregular  
Hours per Week: 2 (lectures)  
Credits: 3  
Preconditions: Knowledge in the Theory of Formal Languages and in the Theory of Programming Languages  
Elaboration/Examination: Individual exercises; exam possible  
Language: German  
Goals: Advanced knowledge in Attribute Grammars and their applications  
Description: Kinds of Attribute Grammars; Related formalisms; Evaluation algorithms and classification; Applications  
Recommended Readings: Announced at the beginning of the lecture

Course: Complex Software Systems - Compiler Compiler  
Lecturer/Instructor: Prof. Riedewald  
Term: Winter semester; irregular  
Hours per Week: 2 (lectures); about 2 hours project work  
Credits: 6  
Preconditions: Programming experiences in a higher programming Language, basic knowledge in compiler techniques  
Elaboration/Examination: Evaluated project  
Language: German  
Goals: Introduction to architecture and application of compiler compilers  
Description: Architecture of compiler compilers; Structure of selected compiler compilers; Applications; Future trends; Project  
Recommended Readings: Announced at the beginning of the lecture

Course: Concepts of Programming Languages  
Lecturer/Instructor: Prof. Riedewald  
Term: Winter semester; irregular  
Hours per Week: 2 (lectures)  
Credits: 3  
Preconditions: Programming experience in a higher programming Language  
Elaboration/Examination: Exam possible  
Language: German  
Goals: Common Language concepts of imperative programming Languages  
Description: Programming Languages as an important means of Software Engineering; History and further development; Evaluation; Common programming Language constructs  
Recommended Readings: Announced at the beginning of the lecture
Course: Object-Oriented Programming Languages
Lecturer/Instructor: Prof. Riedewald
Term: Every Summer semester
Hours per Week: 2 (lectures); project possible
Credits: 3, (6 possible)
Preconditions: Knowledge in ADTs, programming experience in a higher programming Language
Elaboration/Examination: Exam possible
Language: German
Goals: Introduction to main concepts of object-oriented programming Languages
Description: ADT as departure point; Common constructs of object-oriented programming Languages and their implementation in realistic object-oriented programming Languages; Particular Language constructs; Open problems
Recommended Readings: Announced at the beginning of the lecture

Software Engineering

Course: Object-Oriented Software Development
Lecturer/Instructor: Prof. Forbrig
Term: Every Summer semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Algorithms and Data Structures, Software Engineering
Elaboration/Examination: Part of the exam in Software Engineering
Language: German
Goals: Having a deeper look into the object-oriented development of software. Learning good design from examples and patterns.
Description: Principles of object-oriented software development; Examples using Eiffel; Introduction into patterns; UML; Examples using Java; Java and patterns
Recommended Readings: Announced at the beginning of the lecture

Course: Tools for Software Development
Lecturer/Instructor: Prof. Forbrig
Term: Every Summer semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Algorithms and Data Structures, Software Engineering, Object-Oriented Software Development.
Elaboration/Examination: Part of the exam in Software Engineering
Language: German
Goals: Students should be able to use methodologies and the tools which support them
Description: Introduction into the architecture of case tools; Structured methodologies and case/4/0; Object-oriented approaches and objectif, Rational Rose, Delphi, ...
Recommended Readings: Announced at the beginning of the lecture

Course: Requirements Engineering
Lecturer/Instructor: Prof. Forbrig
Term: Every Summer semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Algorithms and Data Structures, Software Engineering
Elaboration/Examination: Part of the exam in Software Engineering
Language: German
Goals: Introduction into the first stage of the software life cycle, informal, semi-formal, formal descriptions of tasks, Learning by doing
Description: Introduction into task analysis; Techniques for the collection of requirements (questionnaires, interviews, video capturing, ...); Techniques for the representation of requirements; Structured and object-oriented approaches (use cases, nets, automata, ...); Specifications using Z
Recommended Readings: Announced at the beginning of the lecture

-------------

Cours de: Langages Orientés-Objet
Lecteur/Instructeur: Prof. Riedewald
Semestre: Semestre d'été
Heures par Semaine: 2 (cours); projet possible
Crédits: 3, (6 possible)
Prérequis: Connaissances en ADTs, expérience de programmation dans un langage de programmation plus élevé
Elaboration/Examen: Possible d'examen
Langue: Allemand
Objectifs: Introduction aux concepts principaux de programmation orientée objet
Description: ADT comme point de départ; Composants communs de la programmation orientée objet et leur implémentation dans une programmation orientée objet réaliste; Constructeurs spécifiques de la langue; Problèmes ouverts
Lectures recommandées: Annulé au début du cours

Système de programmation orientée objet

Course: Concept et langages orientés objet
Lecteur/Instructeur: Prof. Forbrig
Semestre: Semestre d'été
Heures par Semaine: 2 (cours); 2 (exercices)
Crédits: 4
Prérequis: Algorithmes et structures de données, génie logiciel
Elaboration/Examen: Partie de l'examen en génie logiciel
Langue: Allemand
Objectifs: Étude approfondie du développement orienté objet de logiciels. Apprendre le design d'exemples et de modèles.
Description: Principes de développement orienté objet; Exemples utilisant Eiffel; Introduction à des modèles; UML; Exemples utilisant Java; Java et des modèles.
Lectures recommandées: Annulé au début du cours

Système de programmation orientée objet

Course: Tools for Software Development
Lecturer/Instructor: Prof. Forbrig
Semestre: Semestre d'été
Heures par Semaine: 2 (cours); 2 (exercices)
Crédits: 4
Prérequis: Algorithmes et structures de données, génie logiciel, génie logiciel orienté objet
Elaboration/Examen: Partie de l'examen en génie logiciel
Langue: Allemand
Objectifs: Les étudiants doivent être en mesure d'utiliser les méthodologies et les outils qui les soutiennent.
Description: Introduction à l'architecture des outils de cas; Méthodologies structurées et cas/4/0; Approches orientées objet et objectif, Rational Rose, Delphi, ...
Lectures recommandées: Annulé au début du cours

Course: Requirements Engineering
Lecturer/Instructor: Prof. Forbrig
Semestre: Semestre d'été
Heures par Semaine: 2 (cours); 2 (exercices)
Crédits: 4
Prérequis: Algorithmes et structures de données, génie logiciel
Elaboration/Examen: Partie de l'examen en génie logiciel
Langue: Allemand
Objectifs: Introduction à la première étape du cycle de vie du logiciel, informel, semi-formel, formel descriptions de tâches, Learning by doing
Description: Introduction à l'analyse de tâche; Techniques pour la collecte des exigences (questionnaires, interviews, enregistrement vidéo, ...); Techniques pour la représentation des exigences; Approches structurées et orientées objet (cas, états, ...); Spécifications en utilisant Z.
Lectures recommandées: Annulé au début du cours

-------------
Course: Unified Modeling Language (UML)
Lecturer/Instructor: Prof. Forbrig
Term: Every Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Algorithms and Data Structures, Software Engineering
Elaboration/Examination: Part of the exam in Software Engineering
Language: German
Goals: Introduction into object-oriented analysis and design using UML
Description: Introduction into the basic principles of OO, Concepts of UML, Different Notations, Views on models, Transformations, Patterns, Pattern-oriented software development, different approacher for the requirements analysis
Recommended Readings: Announced at the beginning of the lecture

Modeling and Simulation of Computer Systems

Course: Modeling and Simulation of Continuous Systems
Lecturer/Instructor: Prof. Uhrmacher
Term: Every other Summer or Winter semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom, Modeling and Simulation I
Elaboration/Examination: Oral examination (on demand)
Language: German or English / Slides in English
Goals: Introduction to basic concepts of continuous systems simulation
Description: Differential equations; Algebraic Loops; Singularities; Linear and non-linear systems; Characteristic behavior of continuous systems: stable, cyclic, chaotic; Modeling Formalisms: Block Diagrams; Bond Graphs; Systems Dynamics; Simulation Mechanisms: Approximation accuracy; Numerical Stability; Numerical Integration Methods: Runge Kutta Algorithms; Modeling and Simulation systems: modular-hierarchical model construction, separation of model description and experiment description, the example Modelica. Hybrid, discrete continuous models: a system theoretic approach

Course: Parallel and Distributed Modelling and Simulation
Lecturer/Instructor: Prof. Uhrmacher
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom, Modeling and Simulation I
Elaboration/Examination: Oral examination (on demand)
Language: German or English / Slides in English
Goals: Overview of State of the Art Methods in Distributed, Parallel Simulation of Discrete Event Systems
Description: Discrete Event Simulation Fundamentals; Conservative Synchronization Mechanisms; Time Warp; Advanced Optimistic Techniques; Time Parallel Simulation; Distributed Virtual Environments; Networking and Data Distribution; Time Management and Event Ordering; High Level Architecture (HLA)
Course: Agent-Oriented Modeling and Simulation
Lecturer/Instructor: Prof. Uhrmacher
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vor diplom, Modeling and Simulation I
Elaboration/Examination: Oral examination (on demand)
Language: German or English (on demand) / Slides in English
Goals: To purvey an insight into the multi-faceted interplay of software agents and simulation, its methods, theoretical background, and applications.
Description: Relations between Software Agents and Simulation; Simulation of Multi-agent Systems: From Micro Simulation, Object-Oriented Simulation to Agent-Based Simulation; Multi-agent Systems as Dynamic Systems: A system theoretical Approach; Applications in Social Science; Simulation and Testing of Multiagent Systems in Dynamic, Virtual environments: Requirements, Problems and Promises; Testing in Virtual Dynamic, Virtual Environments as Part of the Software Engineering Process; Testing of Artificial Intelligence Agents; Testing of Mobile Agents; Software Agents to construct Simulation Systems: On-Demand Retrieval of Data over the Net; Distributed Execution of Simulation by Communities of Agents; HLA - the High Level Architecture
Recommended Readings: None

------------------------------------------------------------------------------------------------------------------------------------
Course: Methods of Applied Artificial Intelligence
Lecturer/Instructor: Prof. Uhrmacher
Term: Every Summer semester
Hours per Week: 3 (lectures) + 1 (exercise)
Credits: 6
Preconditions: Vor diplom, Modeling and Simulation I
Elaboration/Examination: Oral examination (on demand)
Language: German or English (on demand) / Slides partly in English
Goals: Basic Methods of Artificial Intelligence and their Application
Description: Informed, Non-Informed Search; Genetic Algorithms and genetic programming; Games; Knowledge Representation; Knowledge Acquisition; Fuzzy Systems; Terminological Logic; Planning; Machine Learning; Bayesian Networks; Case-Based Reasoning.

------------------------------------------------------------------------------------------------------------------------------------
Course: Complex Software Systems
Lecturer/Instructor: Prof. Uhrmacher and others
Term: Every other Summer or Winter semester
Hours per Week: 4 (including project work during the entire semester)
Credits: 6
Preconditions: Vor diplom, Modeling and Simulation I
Elaboration/Examination: Individual exercises; project work
Language: German or English (on demand)
Goals: Doing a full software development process for a small modelling and simulation application, from analysis and design up to implementation and documentation, software development in teams of two to four.
The specific topics change every year; (for more detail see the institute’s www page).
Recommended Readings: None

------------------------------------------------------------------------------------------------------------------------------------
Database and Information Systems

Course: Fundamentals of Databases II
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every Summer semester
Hours per Week: 3 (lectures); 1 (exercise)
Credits: 6
Preconditions: Fundamentals of Databases I
Elaboration/Examination: Individual exercises, final examination on request
Language: German
Goals: Learning about implementation techniques for database management systems
Description: File organization and access paths; Query evaluation and optimization; Transaction management; Recovery and data security; Data dictionary, databases and the WWW; Data warehouses; Data mining; Introduction to distributed databases; Extendable databases; Heterogeneous databases; Knowledge bases

Course: Computer Science VI (Database Systems for non-computer scientists)
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every Winter semester
Hours per Week: 2 (lectures); 2(exercises)
Credits: 4
Preconditions: Computer Science I
Elaboration/Examination: Final examinations
Language: German
Goals: Introduction to Fundamentals of Databases I and II, Database Application Programming
Description: This is a compact, introductory version of the topics presented in the lectures "Fundamentals of Databases I"; "Fundamentals of Databases II"; "Database Application Programming"; see there for a list of contents

Course: Active Databases and Deductive Databases
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Fundamentals of Databases I
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding active database systems, deductive databases and knowledge bases
Description: Motivation and basic definitions; Deductive databases (prolog and relational databases, DATALOG, semantics of rule-based Languages, DATALOG optimization and evaluation, prototypes); Active databases (triggers, ECA rules, architectures for active database systems, Termination, confluence, coupling modes, prototypes)
<table>
<thead>
<tr>
<th>Course</th>
<th>Database Application Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Andreas Heuer or Dr. Holger Meyer</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>2 (lecture); 1 (exercise)</td>
</tr>
<tr>
<td>Credits</td>
<td>4</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Fundamentals of Databases I</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Final examination on request</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Introduction to database programming techniques; Database programming</td>
</tr>
<tr>
<td></td>
<td>Languages and systems</td>
</tr>
<tr>
<td>Description</td>
<td>Embedding techniques; Call-level interfaces; Embedded SQL (ESQL);</td>
</tr>
<tr>
<td></td>
<td>Specific versions of ESQ; ESQ in C; Dynamic SQL; Fourth-generation</td>
</tr>
<tr>
<td></td>
<td>Languages; Database programming Languages; Database programming for</td>
</tr>
<tr>
<td></td>
<td>hierarchical and network</td>
</tr>
<tr>
<td></td>
<td>database systems</td>
</tr>
<tr>
<td>Recommended Readings</td>
<td>Heuer, A.; Saake, G.: Datenbanken --- Konzepte und Languagen, 2nd</td>
</tr>
<tr>
<td></td>
<td>Introduction to Database Systems, Volume I, Addison-Wesley, Reading,</td>
</tr>
<tr>
<td></td>
<td>1995; Date, C. J.; Darwen, H.: A Guide to the SQL Standard, Addison-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Information Systems and Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Andreas Heuer; Dr. Holger Meyer</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Fundamentals of Databases I, Object-Oriented Database Systems</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Final examination on request</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Providing an overview on information systems and services, database-</td>
</tr>
<tr>
<td></td>
<td>related and others, specifically the World Wide Web</td>
</tr>
<tr>
<td>Description</td>
<td>Basic principles and classification of information systems; Knowledge</td>
</tr>
<tr>
<td></td>
<td>representation in full-text information systems; Hyper-text and</td>
</tr>
<tr>
<td></td>
<td>hyper-media systems (WWW, HyperWave); XML; Information retrieval</td>
</tr>
<tr>
<td></td>
<td>systems; Digital libraries; Database-based WWW services; Specific</td>
</tr>
<tr>
<td></td>
<td>systems (dbPerl, O2, Informix, CORBA, Java); Advanced</td>
</tr>
<tr>
<td></td>
<td>techniques; WWW search engines</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Complex Software Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Andreas Heuer; Dr.-Ing. Holger Meyer; others</td>
</tr>
<tr>
<td>Term</td>
<td>Every Summer semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>4 (including project work during the whole semester)</td>
</tr>
<tr>
<td>Credits</td>
<td>6</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Fundamentals of Databases I (Fundamentals of Databases II and</td>
</tr>
<tr>
<td></td>
<td>Object-Oriented Database Systems recommended)</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Individual exercises; project work</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Doing a full software development process for a small database</td>
</tr>
<tr>
<td></td>
<td>application, from analysis and design up to implementation and</td>
</tr>
<tr>
<td></td>
<td>documentation, software development in teams of two to four.</td>
</tr>
<tr>
<td>Description</td>
<td>The specific topics change every year; the 1999 topic is setting up</td>
</tr>
<tr>
<td></td>
<td>a workgroup database system for emails using object-relational</td>
</tr>
<tr>
<td></td>
<td>database systems, including (non-GUI) client software based on the</td>
</tr>
<tr>
<td></td>
<td>IMAP4 protocol.</td>
</tr>
<tr>
<td>Recommended Readings</td>
<td>None</td>
</tr>
</tbody>
</table>
Course: Multimedia Database Systems
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every other Summer semester
Hours per Week: 2 (lectures); 1 (exercise)
Credits: 4
Preconditions: Fundamentals of Databases I
(Recommended: Fundamentals of Databases II, Object-Oriented Database Systems)
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding concepts and implementation techniques of multimedia database management systems
Description: Motivation and problem analysis; Analysis of existing database systems; Multimedia applications and data types; Multimedia devices; Database concepts for multimedia contents; Client/server architectures; Multimedia database systems; Queries on multimedia contents; Transaction management and the internal layer of multimedia database systems; Network support for multimedia

------------------------------------------------------------------------------------------------------------------------

Course: Object-Oriented Database Systems
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every Summer semester
Hours per Week: 3 (lectures); 1 (exercise)
Credits: 6
Preconditions: Fundamentals of Databases I, Object-Oriented Programming Languages
Elaboration/Examination: Individual exercises; final examination on request
Language: German
Goals: Introduction to object-oriented database systems, their concepts and models; Programming object-oriented database systems
Description: Introductory examples; Recapitulation of object-oriented programming Languages and relational database systems, Drawbacks of relational database systems; Concepts of object-oriented database systems, the ODMG standard, specific models, query Languages and database systems

------------------------------------------------------------------------------------------------------------------------

Course: Query Evaluation (Fundamentals of Databases III)
Lecturer/Instructor: Dr.-Ing. Holger Meyer
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Fundamentals of Databases I, Fundamentals of Databases II
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding the query evaluation process
Description: Basic principles of query translation and representation (algebra, calculus, query execution plans); Basic algorithms; Translation and evaluation of complex queries; Query optimization heuristics; Cost-based optimization; Search spaces; Search algorithms (depth-first, Hill Climbing, Simulated Annealing, Genetics Programming); Cost functions and ranking; Advanced techniques for distributed and object-oriented database systems
Recommended Readings: None
Course: Semantic Data Models and Database Schema Design
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every other Summer semester
Hours per Week: 2 (lectures); 1 (exercise)
Credits: 4
Preconditions: Fundamentals of Databases I
Elaboration/Examination: Final examination on request
Language: German
Goals: Principles of semantic database models and database schema design
Description: Basic principles of semantic database model; Extended Entity-Relationship models; Functional models, SDM, IFO, GSM, EXTREM; Principles of database schema design; Preserving information capacity; Logical design (ER model, IFO, EXTREM); Object-oriented schema design; Re-engineering existing database structures

-------------------------------------------------------------

Course: Transaction Processing (Fundamentals of Databases III)
Lecturer/Instructor: Dr.-Ing. Holger Meyer
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Fundamentals of Databases I; Fundamentals of Databases II
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding concepts and techniques of transaction handling
Description: Principles of transactions, database architecture and interfaces; Transaction properties: Nested transactions; Serializability; Concurrency control; Interaction between concurrency control and recovery; Concurrency control techniques (lock protocols, optimistic locking, semantic locking); Recovery techniques (logging, transaction manager, commit protocols); Distributed transaction handling; Transactions in non-standard database systems (long transactions, real-time databases, object-oriented databases); Transaction monitors; Specific systems (DB2, Ingres, Tuxedo, X/Open, DTP)
Recommended Readings: None

-------------------------------------------------------------

Course: Theory of Relational Databases
Lecturer/Instructor: Prof. Andreas Heuer
Term: Every other Winter semester
Hours per Week: 3 (lectures), 1 (exercise)
Credits: 6
Preconditions: Fundamentals of Databases I
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding the theoretical foundations of relational database systems and query Languages
Description: Logical foundations; Database schemata and integrity rules; Dependencies; Database schema properties; Decomposition and synthesis algorithms; Tableau theory; Exact optimization methods; Power of query Languages; Updates on views; Concurrency control theory

-------------------------------------------------------------
Course: Distributed Database Systems and Transactions (Fundamentals of Databases III)
Lecturer/Instructor: Dr.-Ing. Holger Meyer
Term: Every summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Fundamentals of Databases I; Fundamentals of Databases II
Elaboration/Examination: Final examination on request
Language: German
Goals: Understanding the issues of and solutions for distributed database systems and their transaction handling
Description: Basic principles (heterogeneity, autonomy, transparency, consistency); Database schema design for distributed databases; Query evaluation and optimization; Distributed transaction management; Concurrency control and recovery in distributed database systems; Examples of distributed database systems
Recommended Readings: None

Course: Distributed Systems
Lecturer/Instructor: Dr.-Ing. Holger Meyer
Term: Every other Summer semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Operating Systems; Computer Networks
Elaboration/Examination: Final examination on request
Language: German
Goals: Introduction to concepts and implementation techniques for distributed systems
Description: Design principles (heterogeneity, autonomy, transparency, and consistency); Interprocess communication; Client-server models; RPC model; Data-flow machines; Process synchronization; Centralized systems; Distributed algorithms; Message-oriented communication; Message passing; Distributed memory management; Shared virtual memory; Distributed transactions; process management; Distributed processes; Remote execution and process migration; Load distribution and balancing; Specific systems: Mach, Plan9, Amoeba, OSF/DCE, OMG/CORBA
Recommended Readings: None
Computer Engineering

Computer Architecture

Course: Practical Training on Microprocessors
Lecturer/Instructor: D. Tavanarian / F. Burchert
Term: Every Summer semester
Hours per Week: 4
Credits: 6
Preconditions: CS I+II
Elaboration/Examination: Tests
Language: German
Goals: Practical work with microprocessor systems for special applications
Description: Knowledge of practical dealing with microprocessors given in the laboratory; Developing functions for specific fields of application of microprocessors, particularly from the areas of automatic control and embedded systems; Solutions of control problems (real time algorithms, algorithms with close temporal boundaries, memory and power requirements, appropriate program structures and the conception hole total systems); Use of modern microprocessors available on the market for practical applications
Recommended Readings: R. Johanns: Handbuch des 80C166, Siemens AG, Abt. Verlag (1993);
K.-H. Mattheis, S. Storandt: Arbeiten mit C166-Controlern, O. Feger Hard+Software Verlag, Traunreut (1995);

Course: Introduction to Computer Architectures (ERA)
Lecturer/Instructor: D. Tavanarian
Term: Every Summer semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Vordiplom would be useful
Elaboration/Examination: Oral examination
Language: German
Goals: Understanding basic computer architectures and their organization
Description: Conceptions, structures and the behavior of digital computers; Implementation and operation of computing systems
Topics of the lecture: taxonomy and organization of computers, SIMD and MIMD parallel systems, array processors, systolic array architectures, associative (i.e. content-addressable) memory and processor architectures, communication structures in parallel systems etc.
W. Erhard, Rechnerarchitektur: Einführung und Grundlagen, B. Teubner Stuttgart 1995;
H. Kunsemüller: Digitale Rechenanlagen, B.G. Teubner Stuttgart 1998;
Course: Network Based Computing
Lecturer/Instructor: D. Tavangarian
Term: Every Summer semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Vordiplom would be useful
Elaboration/Examination: Oral examination
Language: German
Goals: Understanding network-based distributed computing and network topologies and tools for such applications
Description: With regard to distributed system architectures based on networks of workstations and PCs both in the local-area and wide-area networks, the new Terms, technologies, tools, and standards will be discussed. The Course will present different conceptions to design workstation cluster architectures, novel communication structures, adapted to special fields of application, parallel and distributed workstation cluster systems for high throughput and high performance tasks. We will discuss both theoretically and practically significant methods.

Course: Unconventional Computer Architectures
Lecturer/Instructor: D. Tavangarian
Term: Every Winter semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Vordiplom would be useful
Elaboration/Examination: Oral examination
Language: English
Goals: Understanding application-oriented and special computer architectures
Description: Since the beginnings of computer technology, the challenge of closing the computation gap, i.e. the difference between the computing performance demanded by applications and the computing performance given by computer systems, has created many alternatives to the so-called 'traditional' von Neumann conception. Although these alternatives have proved their effectiveness in practice since the early days of computers, there is still a demand for higher performance due to the explosion of information.

The intention of this Course is to analyze the so-called von Neumann concept from different perspectives and to study some of the recent architectures which violate this concept. The Course is aimed at stimulating research interests in the area of computer architecture. To broaden the scope of the Course, an attempt has been made to make the chapters as independent as possible. Each chapter has its own specific approach.

Course: Simulation and Synthesis of Digital Systems
Lecturer/Instructor: D. Tavangarian
Term: Every Winter semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Vordiplom would be useful
Elaboration/Examination: Oral examination
Language: German
Goals:: Understanding structured design and the necessary tools and algorithms
description: Detailed ideas and algorithms of the design of integrated circuits, computer
components and systems combined with an introduction to modern design tools;
Algorithms and models for the specification, description, simulation, and synthesis of
integrated circuits; Hardware specification Language VHDL; Procedures presented
practically employed in exercises and exemplary designs; Multimedia design and
simulation system on the basis of the hardware description Language VHDL
Recommended Readings: B. Eschermann: Funktionaler Entwurf digitaler Schaltungen, Methoden und CAD-
Techniken, Springer-Verlag (1993);
P. Marwedel: Synthese und Simulation von VLSI-Systemen, Carl Hanser Verlag
(1993);
H.-J. Wunderlich: Hochintelligente Schaltungen: Prüfgerechter Entwurf und Test,

Information and Communication Services

Course: Java and Web Technology
Lecturer/Instructor: Prof. Cap
Term: Every Winter semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Elementary programming knowledge (any Language)
Elaboration/Examination: Oral examination and individual exercises
Language: German
Goals: To be able to design internet services with the Java technology
Description: Introduction into Java technology; Java as an object oriented programming Language;
the Java Virtual Machine; Security concepts of mobile code; Java middleware concepts
(JDBC, EJB, CORBA, etc.); Java appliances; Java based smartcards, Jini and
spontaneous networking in Java; Performance problems and their solution; Design
patterns in Java; Survey on Java APIs and supportive technologies (multimedia,
external devices, XML, etc.)
Recommended Readings: Announced at the beginning of the lecture.
Course: Distributed Systems
Lecturer/Instructor: Prof. Cap
Term: Every Winter semester
Hours per Week: 2 (lectures); 2 (exercises)
Credits: 4
Preconditions: Elementary Java programming knowledge
Elaboration/Examination: Oral examination and individual exercises
Language: German
Goals: To be able to design distributed systems and to evaluate distributed architectures
Description: Client/server; Remote procedure call; Remote method invocation; Message passing; Distributed shared memory concepts; Middleware architecture concepts; Time and state in distributed systems; Taxonomy of distributed architectures; Fault tolerance in distributed systems; Distributed algorithmic concepts; Algorithms for virtual global time; Vector time; Consistent snapshots; Data replication and transactions; Security concepts of distributed systems; Specification techniques
Recommended Readings: Announced at the beginning of the lecture.

<table>
<thead>
<tr>
<th>Course</th>
<th>Planning, management and configuration of local computer networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Cap, Dipl.-Phys. Bülow</td>
</tr>
<tr>
<td>Term:</td>
<td>Summer semester (every second)</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures); 2 (lab session)</td>
</tr>
<tr>
<td>Credits:</td>
<td>4</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Computer network fundamentals</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Oral examination and individual exercises</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>To be able to set up, run and maintain a computer network from the technical as well as from the organizational and economical point of view</td>
</tr>
<tr>
<td>Description:</td>
<td>Communication architecture; Active and passive components of a computer network; Cabling systems, organizing the operation of a computer network; System security, backup procedures, crash recovery techniques, preventive maintenance; Ordering, installing and evaluating components on a systematic basis, dealing with user problems; Help desk management; File system structure; Programming interfaces to a networked system; Migration strategies to new technologies</td>
</tr>
<tr>
<td>Recommended Readings:</td>
<td>Announced at the beginning of the lecture.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Internet Protocols and Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Cap</td>
</tr>
<tr>
<td>Term:</td>
<td>Summer semester (every second)</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures); 2 (lab session)</td>
</tr>
<tr>
<td>Credits:</td>
<td>4</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Computer network fundamentals</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Oral examination and individual exercises</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>To be able to work with contemporary internet protocols</td>
</tr>
<tr>
<td>Description:</td>
<td>Standardisation and RFCs; Description of Internet protocols, a survey on the technical details of popular and important Internet protocols and services (ftp, telnet, http, file service, name service, security services, etc), electronic mail systems, web server protocols; Internet protocol security; Specification techniques for protocols; Quality of service problems in the Internet; Multimedia streams on the Internet; Configuration of servers, clients and protocol gateways; GUI development for Internet services</td>
</tr>
<tr>
<td>Recommended Readings:</td>
<td>Announced at the beginning of the lecture.</td>
</tr>
</tbody>
</table>
Theoretical Computer Science

Course: Symbolic Computation I
Lecturer/Instructor: Prof. Hantzschmann / Prof. Widiger (alternating)
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Verbal examination (Komplexeprüfung)
Language: German
Goals: Introduction to Symbolic Computation
Description: Basic problems of symbolic computation; Algebraic fundamentals; Simple Algorithms; Working with Maple
Recommended Readings: Announced at the beginning of the lecture

Course: Symbolic Computation II
Lecturer/Instructor: Prof. Hantzschmann / Prof. Widiger (alternating)
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Symbolic Computation I
Elaboration/Examination: Verbal examination (Komplexeprüfung)
Language: German
Goals: Advanced algorithms of Symbolic Computation
Description: Modular methods; p-adic methods; Symbolic integration; Solving ODE’s
Recommended Readings: Announced at the beginning of the lecture

Course: Rewriting Systems
Lecturer/Instructor: Prof. Widiger
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Verbal examination (Komplexeprüfung)
Language: German
Goals: Introducing basic concepts of rewriting
Description: Reducing concepts; Terms and algebras; Termination; Unification; Critical pairs; Completion
Recommended Readings: Announced at the beginning of the lecture

Course: Lisp programming
Lecturer/Instructor: Prof. Widiger
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: None
Elaboration/Examination: Programming exercises
Language: German
Goals: Principles of functional programming, programming with Common Lisp
Description: Lisp programming; CLOS; λ-calculus
Recommended Readings: Announced at the beginning of the lecture

Course: Algorithmics
Lecturer/Instructor: Prof. Hantzschmann
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Verbal examination
Language: German
Goals: Mathematical and informatic foundations and methods for development and analysis of algorithms
Description: Mathematical foundations; Advanced design and analysis techniques; NP-complete problems and approximation methods; Parallel algorithms
Recommended Readings: Announced at the beginning of the lecture
Course: Computer Analysis
Lecturer/Instructor: Prof. Hantzschmann
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Verbal examination
Language: German
Goals: Introduction in conception and algorithms of Computer analysis
Description: Concepts and methodology; Approximation of functions; Algorithms for the approximate solution of ordinary differential equations; Practicable error estimations; Application of CAS
Recommended Readings: Announced at the beginning of the lecture

Course: Numerical Computing
Lecturer/Instructor: Prof. Hantzschmann
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Verbal examination
Language: German
Goals: Basic problems of numerical computing
Description: Error analysis of floating-point computation; Numerical stability; Interval arithmetic; Axioms of an optimal computer arithmetic; Automatic result verification; Software systems for scientific computing
Recommended Readings: Announced at the beginning of the lecture

Course: Efficiency of Algorithms and Algorithmic Problems
Lecturer/Instructor: Prof. Brandstädt
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Oral examination (Komplexprüfung)
Language: German
Goals: Introduction to basic concepts and techniques in complexity and efficiency
Description: Efficiency of algorithms on various machine models; Time and space on machine models; Complexity classes; Completeness in various complexity classes and reducibility concepts; DeTerminism versus non-deTerminism; Time versus space; Complexity on parallel machine models; Circuit complexity; Communication complexity in parallel machine models; Structural complexity theory

Course: Efficient Graph Algorithms I
Lecturer/Instructor: Prof. Brandstädt
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Oral examination (Komplexprüfung)
Language: German
Goals: Introduction to basic concepts and techniques in efficient graph algorithms
Description: Basic algorithmic graph problems; Linear time node and edge searching methods for graphs; Depth-first search (DFS) and breadth-first search (BFS); Applications of DFS and BFS; Minimum spanning trees; Greedy algorithm and matroids; Shortest paths; Maximum flow in networks; Steiner tree problem and domination problem
<table>
<thead>
<tr>
<th>Course:</th>
<th>Efficient Graph Algorithms II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Brandstädt</td>
</tr>
<tr>
<td>Term:</td>
<td>irregular</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits:</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Vordiplom</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Oral examination (Komplexprüfung)</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introduction to basic and advanced concepts and techniques in efficient graph algorithms and special classes of graphs</td>
</tr>
<tr>
<td>Description:</td>
<td>Maximum flow and maximum matching; Coloring nodes and edges in graphs; Generalized tree structure and its algorithmic use; Some basic graph classes; Robust algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course:</th>
<th>Approximation Algorithms and Approximation Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Dr. Le; Prof. Brandstädt</td>
</tr>
<tr>
<td>Term:</td>
<td>irregular</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits:</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Vordiplom</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Oral examination (Komplexprüfung)</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introduction to basic concepts and techniques in approximation algorithms and complexity of approximation problems</td>
</tr>
<tr>
<td>Description:</td>
<td>Data structures and complexity classes; Optimization problems and approximation algorithms; Well-approximable problems; Polynomial time approximation schemes (PTAS): Job-shop-scheduling; Knapsack problem; Travelling Salesperson Problem (TSP); Steiner tree problem; Linear optimization and LP relaxation; Semi-definite programming; Problems which are hard to approximate and methods to show this</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course:</th>
<th>Foundations of Cryptography</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Brandstädt</td>
</tr>
<tr>
<td>Term:</td>
<td>irregular</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits:</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Vordiplom</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Oral examination (Komplexprüfung)</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introduction to basic concepts and techniques in cryptography with emphasis on public-key cryptography and RSA, number-theoretic tools on which RSA is based</td>
</tr>
<tr>
<td>Description:</td>
<td>Classic encryption and decryption; Idea of public key methods; Basic number theory: Euler's OE function, large prime numbers, Euler's theorem; Rivest-Shamir-Adleman (RSA) algorithm; Other public key methods; Security problems; Signatures; Protocols</td>
</tr>
</tbody>
</table>
Course: Graph and Hypergraph Models in Computer Science
Lecturer/Instructor: Prof. Brandstädt
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Oral examination (Komplexprüfung)
Language: German
Goals: Introduction to basic concepts and techniques for graph models in various fields of computer science
Description: Graph models for parallel processor networks; Acyclic hypergraphs, chordal graphs and relational database schemes; VLSI design and gate-matrix layout; Graphs, posets and scheduling; Interval graphs, Special graph classes with linear structure; Human genome project

--------------------------------------------------------
Course: Foundations of Parallel Computers and Parallel Algorithms
Lecturer/Instructor: Prof. Brandstädt
Term: irregular
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Vordiplom
Elaboration/Examination: Oral examination (Komplexprüfung)
Language: German
Goals: Introduction to basic concepts and techniques in parallel computing
Description: Parallel computer models; Basic parallel processor networks; Efficient parallel algorithms; Routing and communication in parallel processor networks; Problems which are hard to parallelize and the complexity class NC

--------------------------------------------------------
Computer Graphics

Course: Computer Animation
Lecturer/Instructor: Prof. Jackèl
Term: Every Summer semester
Hours per Week: 2 (lectures); project “Complex Software Systems” possible
Credits: 3, (4 possible)
Preconditions: Computer Graphics I
Elaboration/Examination: Verbal examination
Language: German
Goals: Introduction into the principles of three-dimensional computer Animation
Description: Keyframing techniques; Quaternions; Camera animation; Animating deformation; Metaballs; Particle Systems; Direct and inverse kinematics; Facial animation; Physically based modelling and animation; Collision detection.
Recommended Readings: A. Watt; M. Watt: Advanced animation and rendering techniques, Addison-Wesley.

Course: Computer Vision I
Lecturer/Instructor: Prof. Jackèl
Term: Every Summer semester
Hours per Week: 2 (lectures), 1 (exercise)
Credits: 4
Preconditions: None
Elaboration/Examination: Verbal examination
Language: German
Goals: Introduction into the fundamentals and principles of Image Processing
Description: Fourier Transform; Filtering and filter design; Segmentation; Image Enhancement; Processing of binary images; Image Compression; Applications.

Course: Computer Vision II
Lecturer/Instructor: Prof. Jackèl
Term: Every Winter semester
Hours per Week: 1 (lecture), 1 (exercise)
Credits: 2
Preconditions: Computer Vision I
Elaboration/Examination: Verbal examination
Language: German
Goals: Introduction into principles of image analysis and image recognition
Description: Image restoration and Image reconstruction; Feature extraction; Neural Networks; Stereo Vision, Shape from X techniques

Course: Computer Graphics and Multimedia Hardware
Lecturer/Instructor: Prof. Jackèl
Term: Every Winter semester
Hours per Week: 2 (lectures)
Credits: 3
Preconditions: Computer Graphics I
Elaboration/Examination: Verbal examination
Language: German
Goals: Introduction into the fundamentals and principles of computer graphics multimedia hardware
Description: Input and output devices; Principles raster display hardware; Flat-panel displays; Stereo and volume displays; Real-Time-Displays; Virtual reality an virtual environments
<table>
<thead>
<tr>
<th>Course:</th>
<th>Geometric Modelling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Schumann</td>
</tr>
<tr>
<td>Term:</td>
<td>Every Summer semester</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures); project “complex software systems” possible</td>
</tr>
<tr>
<td>Credits:</td>
<td>3 (4 possible)</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Computer Graphics I</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Verbal examination</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introducing basic concepts in solid- and surface modelling</td>
</tr>
<tr>
<td>Description:</td>
<td>Basic concepts of solid- and surface modelling; Functionality of CAD and geometric systems</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course:</th>
<th>Scientific Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer/Instructor:</td>
<td>Prof. Schumann</td>
</tr>
<tr>
<td>Term:</td>
<td>Every Summer semester</td>
</tr>
<tr>
<td>Hours per Week:</td>
<td>2 (lectures); project “complex software systems” possible</td>
</tr>
<tr>
<td>Credits:</td>
<td>3 (4 possible)</td>
</tr>
<tr>
<td>Preconditions:</td>
<td>Computer Graphics I</td>
</tr>
<tr>
<td>Elaboration/Examination:</td>
<td>Verbal examination</td>
</tr>
<tr>
<td>Language:</td>
<td>German</td>
</tr>
<tr>
<td>Goals:</td>
<td>Introduction of important concepts for visualizing scientific, environmental, medical and engineering data</td>
</tr>
<tr>
<td>Description:</td>
<td>Concepts for visualizing scientific, environmental, medical and engineering data; Multiparameter data in space and time; Visualizing volume and flow data</td>
</tr>
<tr>
<td>Course</td>
<td>Aspects of Computer Graphics</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Lecturer/Instructor</td>
<td>Prof. Schumann</td>
</tr>
<tr>
<td>Term</td>
<td>Every Winter semester</td>
</tr>
<tr>
<td>Hours per Week</td>
<td>2 (lectures)</td>
</tr>
<tr>
<td>Credits</td>
<td>3</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Computer Graphics I</td>
</tr>
<tr>
<td>Elaboration/Examination</td>
<td>Verbal examination</td>
</tr>
<tr>
<td>Language</td>
<td>German</td>
</tr>
<tr>
<td>Goals</td>
<td>Overview and state-of-the-art in current research projects and applications of computer graphics</td>
</tr>
<tr>
<td>Description</td>
<td>Changes, depends on current topics</td>
</tr>
<tr>
<td>Recommended Readings</td>
<td>Announced at the beginning of the lecture</td>
</tr>
</tbody>
</table>

| Course                          | Multimedia Communication     |
| Lecturer/Instructor             | Prof. Urban                 |
| Term                           | Every Winter semester        |
| Hours per Week                  | 2 (lectures)                 |
| Credits                        | 3                             |
| Preconditions                  | Computer Graphics I           |
| Elaboration/Examination         | Verbal examination            |
| Language                       | German                        |
| Goals                          | Introduction into different aspects of multimedia communication |
| Description                    | Introduction and Terms; Standards in multimedia; Multimedia in education and training; Course management system/modular training system; CSCW; Transfer systems; Multimedia authoring systems |
| Recommended Readings           | Announced at the beginning of the lecture |

| Course                          | Object-Oriented Programming and Computer Graphics |
| Lecturer/Instructor             | Dr. Karstens                 |
| Term                           | Every Summer semester        |
| Hours per Week                  | 2 (lectures)                 |
| Credits                        | 3                             |
| Preconditions                  | Computer Graphics I           |
| Elaboration/Examination         | Verbal examination            |
| Language                       | German                        |
| Goals                          | Overview over concepts in object-oriented programming by examples of computer graphics and user interface design |
| Description                    | Concepts of Smalltalk, C++ and Java for graphical programming; User interfaces with Smalltalk; Object-oriented class libraries; Design patterns for user interface design; Open Inventor Toolkit and VRML and Java 3D |

| Course                          | Dialog Systems and Software Ergonomics |
| Lecturer/Instructor             | Dr. Karstens                 |
| Term                           | Every Summer semester        |
| Hours per Week                  | 2 (lectures)                 |
| Credits                        | 3                             |
| Preconditions                  | Computer Graphics I           |
| Elaboration/Examination         | Verbal examination            |
| Language                       | German                        |
| Goals                          | Introduction into technical and ergonomical aspects of user interface design |
| Description                    | Design principles; Human factors; Guidelines and standards |