MODULDETAILS ÜBERSICHT

ANALYSIS I

Kurzbeschreibung:

* Reelle und komplexe Zahlen
* Grenzwerte
* Folgen
* Reihen
* Potenzreihen
* stetige Abbildungen
* Differential- und Integralrechnung einer Variablen
* Einführung in gewöhnliche Differentialgleichungen

Lernziel: Einführung in die Grundlagen der Analysis

ANALYSIS II

Kurzbeschreibung: Einführung in die mehrdimensionale Differential- und Integralrechung.

Lernziel: Einführung in die Grundlagen der Analysis

Inhalt: • Differenzierbare Abbildungen • Maxima und Minima • der Satz über implizite Funktionen • mehrfache Integrale • Integration über Untermannigfaltigkeiten • die Sätze von Gauss und Stokes

ANALYSIS III

Kurzbeschreibung: In this lecture we treat problems in applied analysis. The focus lies on the solution of quasilinear first order PDEs with the method of characteristics, and on the study of three fundamental types of partial differential equations of second order: • the Laplace equation • the heat equation • the wave equation

Lernziel: The aim of this class is to provide students with a general overview of first and second order PDEs, and teach them how to solve some of these equations using characteristics and/or separation of variables.

Inhalt:

1. General introduction to PDEs and their classification • linear • quasilinear • semilinear • nonlinear • elliptic • parabolic • hyperbolic
2. Quasilinear first order PDEs • Solution with the method of characteristics • Conservation laws
3. Hyperbolic PDEs • wave equation • d'Alembert formula in (1+1)-dimensions • method of separation of variables
4. Parabolic PDEs • heat equation • maximum principle • method of separation of variables
5. Elliptic PDEs • Laplace equation • maximum principle • method of separation of variables • variational method

PHYSIK I

Kurzbeschreibung: Einführung in die Denk- und Arbeitsweise in der Physik unter Zuhilfenahme von Demonstrationsexperimenten: • Mechanik von Massenpunkten und starren Körpern • Schwingungen und Wellen

Lernziel: Vermittlung der physikalischen Denk- und Arbeitsweise und Einführung in die Methoden in einer experimentellen Wissenschaft. Die Studenten sollen lernen, physikalische Fragestellungen im eigenen Wissenschaftsbereich zu identifizieren, zu kommunizieren und zu lösen.

Inhalt: • Mechanik (Bewegung, Newtonsche Axiome, Arbeit und Energie, Impulserhaltung, Drehbewegungen, Gravitation, deformierbare Körper) • Schwingungen und Wellen (Schwingungen, mechanische Wellen, Akustik)

PHYSIK II

Kurzbeschreibung: Einführung in die Denk- und Arbeitsweise in der Physik unter Zuhilfenahme von Demonstrationsexperimenten: • Elektrizität und Magnetismus • Licht • Einführung in die Moderne Physik

Lernziel: Vermittlung der physikalischen Denk- und Arbeitsweise und Einführung in die Methoden in einer experimentellen Wissenschaft. Der Student/in soll lernen physikalische Fragestellungen im eigenen Wissenschaftsbereich zu identifizieren, zu kommunizieren und zu lösen.

KOMPLEXE ANALYSIS

Kurzbeschreibung: • Grundlagen der Komplexen Analysis in Theorie und Anwendung, insbesondere globale Eigenschaften analytischer Funktionen. • Einführung in die Integraltransformationen und Beschreibung einiger Anwendungen.

Lernziel: Erwerb von einigen grundlegenden Werkzeugen der komplexen Analysis, sowie Verständnis und Anwendung Fourier- und Laplacetransformationen.

Inhalt: • Beispiele analytischer Funktionen • Cauchyscher Integralsatz • Taylor- und Laurententwicklungen • Singularitäten analytischer Funktionen • Residuenkalkül • Fourierreihen und Fourier-Transformation • Laplace-Transformation

CHEMIE

Kurzbeschreibung: Einführung in die Chemie mit Aspekten aus der anorganischen, organischen und physikalischen Chemie.

Lernziel: • Einfache Modelle der chemischen Bindung und der dreidimensionalen Struktur von Molekülen verstehen • Ausgewählte chemische Systeme anhand von Reaktionsgleichungen und Gleichgewichtsrechnungen beschreiben und quantitativ erfassen • Grundlegende Begriffe der chemischen Kinetik (z. B. Reaktionsordnung, Geschwindigkeitsgesetz und -konstante) verstehen und anwenden.

Inhalt: • Periodisches System der Elemente • chemische Bindung (LCAO-MO) • molekulare Struktur (VSEPR) • Reaktionen • Gleichgewicht • chemische Kinetik.

DATENSTRUKTUREN UND ALGORITHMEN

Kurzbeschreibung: • Der Kurs vermittelt die Grundlagen für den Entwurf und die Analyse von Algorithmen. • Anhand klassischer Probleme werden gängige Datenstrukturen, Algorithmen und Paradigmen für den Algorithmenentwurf diskutiert. • Der Kurs umfasst auch eine Einführung in die parallele und nebenläufige Programmierung und das Programmiermodell von C++ wird eingehend diskutiert.

Lernziel: Verständnis des Entwurfs und der Analyse grundlegender Algorithmen und Datenstrukturen. Wissen um die Chancen, Probleme und Grenzen der parallelen und nebenläufigen Programmierung. Vertiefter Einblick in ein modernes Programmiermodell anhand der Programmiersprache C++.

Inhalt: Datenstrukturen und Algorithmen: • Mathematische Tools für die Analyse von Algorithmen (asymptotisches Funktionenwachstum, Rekursionsgleichungen, Rekursionsbäume) • informelle Beweise für die Korrektheit von Algorithmen (Invarianten und Codetransformation) • Entwurfsparadigmen für die Entwicklung von Algorithmen (Induktion, Divide-and-Conquer, Sweep-Line-Methode, Backtracking und dynamische Programmierung) • klassische algorithmische Probleme (Suche, Auswahl und Sortierung), Datenstrukturen für verschiedene Zwecke (verkettete Listen, Hash-Tabellen, balancierte Suchbäume, Quad-Trees, Heaps, Union-Find) • weitere Tools für die Laufzeitanalyse (z.B. amortisierte Analyse)

Die Beziehung und enge Kopplung zwischen Algorithmen und Datenstrukturen wird anhand von geometrischen Problemen (konvexe Hülle, Linienschnitte, dichteste Punktepaare) und Graphenalgorithmen (Traversierungen, topologische Sortierung, transitive Hülle, kürzeste Pfade, minimale Spannbäume, maximaler Fluss) illustriert.

Programmiermodell von C++: • korrekte und effiziente Speicherbehandlung • generische Programmierung mit Templates • funktionale Ansätze mit Funktoren und Lambda-Ausdrücken

Parallele Programmierung: • Konzepte der parallelen Programmierung (Amdahl/Gustavson, Task/Daten-Parallelität, Scheduling) • Probleme der Nebenläufigkeit (data races, bad interleavings, memory reordering) • Prozess-Synchronisation und Kommunikation in einem Shared-Memory-System (Mutual Exclusion, Semaphoren, Monitore, Condition-Variablen) • Fortschrittsbedingungen (Deadlock-Freiheit, Starvation).

INFORMATIK

Kurzbeschreibung: Die Vorlesung bietet eine Einführung in das Programmieren mit einem Fokus auf systematischem algorithmischem Problemlösen. Lehrsprache ist C++. Es wird keine Programmiererfahrung vorausgesetzt.

Lernziel: Primäres Lernziel der Vorlesung ist die Befähigung zum Programmieren mit C++. Studenten beherrschen nach erfolgreichem Abschluss der Vorlesung die Mechanismen zum Erstellen eines Programms. Sie kennen • die fundamentalen Kontrollstrukturen • Datenstrukturen • verstehen wie man ein algorithmisches Problem in ein Programm abbildet. Sie haben eine Vorstellung davon, was "hinter den Kulissen" passiert, wenn ein Programm übersetzt und ausgeführt wird.

Sekundäre Lernziele der Vorlesung sind • das Computer-basierte, algorithmische Denken • Verständnis der Möglichkeiten und der Grenzen der Programmierung • die Vermittlung der Denkart eines Computerwissenschaftlers

Inhalt: Wir behandeln • fundamentale Datentypen • Ausdrücke und Anweisungen • (Grenzen der) Computerarithmetik • Kontrollanweisungen • Funktionen • Felder • zusammengesetzte Strukturen • Zeiger Im Teil zur Objektorientierung werden Klassen, Vererbung und Polymorphie behandelt, es werden exemplarisch einfache dynamische Datentypen eingeführt. Die Konzepte der Vorlesung werden jeweils durch Algorithmen und Anwendungen motiviert und illustriert.

DISKRETE MATHEMATIK

Kurzbeschreibung: Inhalt: • Mathematisches Denken und Beweise • Abstraktion • Mengen • Relationen (z.B. Äquivalenz- und Ordnungsrelationen) • Funktionen • (Un-)abzählbarkeit • Zahlentheorie • Algebra (Gruppen, Ringe, Körper, Polynome, Unteralgebren, Morphismen) • Logik (Aussagen- und Prädikatenlogik, Beweiskalküle).

Lernziel: Hauptziele der Vorlesung sind • die Einführung der wichtigsten Grundbegriffe der diskreten Mathematik • das Verständnis der Rolle von Abstraktion und von Beweisen • die Diskussion einiger Anwendungen, z.B. aus der Kryptographie, Codierungstheorie und Algorithmentheorie

LINEARE ALGEBRA

Kurzbeschreibung: Inhalt: • Lineare Gleichungssysteme - der Algorithmus von Gauss • Matrizen - LR-Zerlegung • Determinanten • Vektorräume • Ausgleichsrechnung - QR-Zerlegung • Lineare Abbildungen • Eigenwertproblem • Normalformen - Singulärwertzerlegung; numerische Aspekte

Lernziel: Sie können: • lineare Gleichungssysteme mit dem Gauss-Algorithmus lösen; • die Grundoperationen mit Vektoren und Matrizen anwenden und geometrisch interpretieren; • verschiedene Matrizenzerlegungen berechnen und passend anwenden; • Eigenwerte, Eigenvektoren und Determinanten von Matrizen berechnen; • Eigenschaften von Vektorräumen und linearen Abbildungen verwenden; • lineare Ausgleichsprobleme formulieren und mit geeigneten Methoden lösen; • die Singulärwertzerlegung verstehen und anwenden.

Inhalt: • Lineare Gleichungssysteme, Matrizen, Gauss-Elimination, LU- und QR-Zerlegungen • Lineare Räume, Fundamentalsatz der linearen Algebra - Teil I, Basiswahl und Basiswechsel • Lineare Abbildungen und Abbildungsmatrix bei Koordinatentransformationen • Norm und Skalarprodukt in linearen Räumen, Gram-Schmidt-Algorithmus, Projektoren • Lineare Ausgleichsrechnung • Determinanten • Eigenwerte und Eigenvektoren, Symmetrische Matrizen • Singulärwertzerlegung und Fundamentalsatz der linearen Algebra, Anwendungen

NUMERICAL METHODS FOR PARTIAL DIFFERENTIAL EQUATIONS

Kurzbeschreibung: Derivation, properties, and implementation of fundamental numerical methods for a few key partial differential equations: • convection-diffusion • heat equation • wave equation • conservation laws Implementation in C++ based on a finite element library.

Lernziel: Main skills to be acquired in this course: • Ability to implement fundamental numerical methods for the solution of partial differential equations efficiently. • Ability to modify and adapt numerical algorithms guided by awareness of their mathematical foundations. • Ability to select and assess numerical methods in light of the predictions of theory • Ability to identify features of a PDE (= partial differential equation) based model that are relevant for the selection and performance of a numerical algorithm. • Ability to understand research publications on theoretical and practical aspects of numerical methods for partial differential equations. • Skills in the efficient implementation of finite element methods on unstructured meshes. This course is neither a course on the mathematical foundations and numerical analysis of methods nor a course that merely teaches recipes and how to apply software packages.

Inhalt: • Second-order scalar elliptic boundary value problems • Finite-element methods (FEM) • FEM: Convergence and Accuracy • Beyond FEM: Alternative Discretizations • Non-linear elliptic boundary value problems • Second-order linear evolution problems • Convection-diffusion problems • Finite Elements for the Stokes Equation

STOCHASTIK

Kurzbeschreibung: Wahrscheinlichkeitsmodelle und Anwendungen, Einführung in Schätztheorie und statistische Testtheorie

Lernziel: Fähigkeit, die behandelten wahrscheinlichkeitstheoretischen Methoden und Modelle zu verstehen und anzuwenden. Fähigkeit, einfache statistische Tests selbst durchzuführen und die Resultate zu interpretieren.

Inhalt: Der Begriff Wahrscheinlichkeitsraum und einige klassische Modelle: • Axiome von Kolmogorov • einfache Folgerungen • diskrete Modelle • Dichtefunktionen • Produktmodelle • Zusammenhang zwischen den bisher betrachteten Modellen • Verteilungsfunktionen • Transformation von Wahrscheinlichkeitsverteilungen.

Bedingte Wahrscheinlichkeiten: • Definition und Beispiele • Berechnung von absoluten aus bedingten Wahrscheinlichkeiten • Bayes'sche Regel • Anwendung auf Nachrichtenquellen • bedingte Verteilungen • Erwartungswert einer Zufallsvariablen • Varianz • Kovarianz und Korrelation • lineare Prognosen • Gesetz der grossen Zahlen • zentraler Grenzwertsatz

Einführung in die Statistik: • Schätzung von Parametern • Tests

FLUIDDYNAMIK I

Kurzbeschreibung: An introduction to the physical and mathematical foundations of fluid dynamics is given. Topics include: • dimensional analysis • integral and differential conservation laws • inviscid and viscous flows • Navier-Stokes equations • boundary layers • turbulence • potential flows • unsteady flows • aerodynamic concepts • vorticity dynamics • compressible flows

Lernziel: Introduction to the physical and mathematical principles of fluid dynamics. Fundamental terminology/principles and their application to simple problems. Concepts, phenomena and quantitative description of potential, rotational, and one-dimensional compressible flows.

Inhalt: • Dimensional analysis • integral and differential conservation laws • inviscid and viscous flows • Navier-Stokes equations • boundary layers, turbulence • potential flows • unsteady flows • aerodynamic concepts • vorticity dynamics • compressible flows

STATISTISCHE PHYSIK UND COMPUTER SIMULATION

Kurzbeschreibung: • Prinzipien und Anwendungen der statistischen Mechanik und Gleichgewichts-Molekulardynamik • Monte-Carlo-Verfahren • stochastische Dynamik • freien Energie-Rechnung Die Übungen verwenden ein Computersimulationsprogramm, um Ensembles zu generieren und danach Ensembledurchschnitte zu berechnen.

Lernziel: • Einführung in die statistische Mechanik mit Hilfe von Computersimulationen • Erwerben der Fertigkeit, Computersimulationen durchzuführen und die Resultate zu interpretieren.

Inhalt: • Prinzipien und Anwendungen der statistischen Mechanik und Gleichgewichts-Molekulardynamik • Monte-Carlo-Verfahren • stochastische Dynamik und freien Energie-Rechnung Die Übungen verwenden ein Computersimulationsprogramm, um Ensembles zu generieren und danach Ensembledurchschnitte zu berechnen.

INTRODUCTION TO MATHEMATICAL OPTIMIZATION

Kurzbeschreibung: Introduction to basic techniques and problems in mathematical optimization, and their applications to a variety of problems in engineering.

Lernziel: The goal of the course is to obtain a good understanding of some of the most fundamental mathematical optimization techniques used to solve linear programs and basic combinatorial optimization problems. The students will also practice applying the learned models to problems in engineering.

Inhalt: Topics covered in this course include: • Linear programming (simplex method, duality theory, shadow prices, ...). • Basic combinatorial optimization problems (spanning trees, shortest paths, network flows, ...). • Modelling with mathematical optimization: applications of mathematical programming in engineering.

NUMERISCHE METHODEN FÜR CSE

Kurzbeschreibung: The course gives an introduction into fundamental techniques and algorithms of numerical mathematics which play a central role in numerical simulations in science and technology. The course focuses on fundamental ideas and algorithmic aspects of numerical methods. The exercises involve actual implementation of numerical methods in C++.

Lernziel: • Knowledge of the fundamental algorithms in numerical mathematics • Knowledge of the essential terms in numerical mathematics and the techniques used for the analysis of numerical algorithms • Ability to choose the appropriate numerical method for concrete problems • Ability to interpret numerical results • Ability to implement numerical algorithms efficiently

Inhalt: • Computing with Matrices and Vectors • Direct Methods for linear systems of equations • Least Squares Techniques • Data Interpolation and Fitting • Iterative Methods for non-linear systems of equations • Filtering Algorithms • Approximation of Functions • Numerical Quadrature

PROGRAMMIERTECHNIKEN FÜR PHYSIKALISCHE SIMULATIONEN

Kurzbeschreibung: This lecture provides an overview of programming techniques for scientific simulations. The focus is on basic and advanced C++ programming techniques and scientific software libraries. Based on an overview over the hardware components of PCs and supercomputer, optimization methods for scientific simulation codes are explained.

Lernziel: The goal of the course is that students learn basic and advanced programming techniques and scientific software libraries as used and applied for scientific simulations.

SYSTEMS PROGRAMMING AND COMPUTER ARCHITECTURE

Kurzbeschreibung: Introduction to systems programming. C and assembly language, floating point arithmetic, basic translation of C into assembler, compiler optimizations, manual optimizations.

How hardware features like superscalar architecture, exceptions and interrupts, caches, virtual memory, multicore processors, devices, and memory systems function and affect correctness, performance, and optimization.

Lernziel: The course objectives are for students to:

1. Develop a deep understanding of, and intuition about, the execution of all the layers (compiler, runtime, OS, etc.) between programs in high-level languages and the underlying hardware: the impact of compiler decisions, the role of the operating system, the effects of hardware on code performance and scalability, etc.
2. Be able to write correct, efficient programs on modern hardware, not only in C but high-level languages as well.
3. Understand Systems Programming as a complement to other disciplines within Computer Science and other forms of software development. This course does not cover how to design or build a processor or computer.

Inhalt:

1. This course provides an overview of "computers" as a platform for the execution of (compiled) computer programs.
2. This course provides a programmer's view of how computer systems execute programs, store information, and communicate.
3. The course introduces the major computer architecture structures that have direct influence on the execution of programs (processors with registers, caches, other levels of the memory hierarchy, supervisor/kernel mode, and I/O structures) and covers implementation and representation issues only to the extend that they are necessary to understand the structure and operation of a computer system.

The course attempts to expose students to the practical issues that affect performance, portability, security, robustness, and extensibility. This course provides a foundation for subsequent courses on operating systems, networks, compilers and many other courses that require an understanding of the system-level issues.

Topics covered include: • machine-level code and its generation by optimizing compilers • address translation • input and output • trap/event handlers • performance evaluation and optimization (with a focus on the practical aspects of data collection and analysis)

SOFTWARE ENGINEERING

Kurzbeschreibung: This course introduces both theoretical and practical aspects of software engineering, all of which are applied in a substantial team project.

Lernziel: The course has two main objectives: • Obtain an end-to-end (both, theoretical and practical) understanding of the core techniques used for building quality software. • Be able to apply these techniques in practice.

Inhalt: This course introduces theoretical and applied aspects of software engineering, including: • requirements • specifications and documentation • formal and informal modelling • modularity • testing and concolic execution.

The theoretical foundations provided in the lecture, from understanding requirements over design and implementation to deployment and change requests, will be applied by the students in a mandatory project that spans the whole semester: developing, as a team, a small multiplayer game with graphical user interface and network support.

Lectures and project use C++, and we expect knowledge corresponding to lecture 252-0856 Computer Science.

DESIGN OF HIGH PERFORMANCE COMPUTING

Kurzbeschreibung: Advanced topics in parallel and high-performance computing.

Lernziel: Understand concurrency paradigms and models from a higher perspective and acquire skills for designing, structuring and developing possibly large parallel high-performance software systems.

Become able to distinguish parallelism in problem space and in machine space. Become familiar with important technical concepts and with concurrency folklore.

Inhalt: We will cover all aspects of high-performance computing ranging from architecture through programming up to algorithms. We will start with a discussion of caches and cache coherence in practical computer systems. We will dive into parallel programming concepts such as memory models, locks, and lock-free. We will cover performance modeling and parallel design principles as well as basic parallel algorithms.

INTRODUCTION INTO MACHINE LEARNING

Kurzbeschreibung: The course introduces the foundations of learning and making predictions based on data.

Lernziel: The course will introduce the foundations of learning and making predictions from data. We will study basic concepts such as trading goodness of fit and model complexity. We will discuss important machine learning algorithms used in practice, and provide hands-on experience in a course project.

Inhalt: • Linear regression (overfitting, cross-validation/bootstrap, model selection, regularization, [stochastic] gradient descent) • Linear classification: Logistic regression (feature selection, sparsity, multi-class) • Kernels and the kernel trick (Properties of kernels; applications to linear and logistic regression); k-nearest neighbor • Neural networks (backpropagation, regularization, convolutional neural networks) • Unsupervised learning (k-means, PCA, neural network autoencoders) • The statistical perspective (regularization as prior; loss as likelihood; learning as MAP inference) • Statistical decision theory (decision making based on statistical models and utility functions) • Discriminative vs. generative modeling (benefits and challenges in modeling joint vs. conditional distributions) • Bayes' classifiers (Naive Bayes, Gaussian Bayes; MLE) • Bayesian approaches to unsupervised learning (Gaussian mixtures, EM)

WISSENSCHAFT IM KONTEXT

Kurzbeschreibung: Wissenschaftliches Arbeiten und Methodenkompetenz

Lernziel: Erlangung von Methodenkompetenzen für wissenschaftliches Arbeiten im Studium und in der beruflichen Praxis

Inhalt: Wissenschaftliche Methoden, Recherche, wissenschaftliches Schreiben

FALLSTUDIEN

Kurzbeschreibung: Praktische Anwendung der im Studium erlernten Fähigkeiten in einem Praxisprojekt

Lernziel: Fähigkeit zum selbständigen Bearbeiten einer Aufgabenstellung aus der Praxis

Inhalt: Projektarbeit in Teams an realen Problemstellungen

BACHELORARBEIT

Kurzbeschreibung: Selbständige Bearbeitung eines wissenschaftlichen Themas

Lernziel: Erlernen und Anwenden wissenschaftlicher Methoden, selbstständiges Bearbeiten einer wissenschaftlichen Fragestellung

Inhalt: Individuell definiertes wissenschaftliches Projekt