

**CORSO DI LAUREA IN INFORMATICA**  
**BACHELOR IN INFORMATIK**

**Contenuto degli insegnamenti**  
**Inhalt der Lehrveranstaltungen**

Analysis <ul style="list-style-type: none"> <li>Sequences and series</li> <li>Univariate functions</li> <li>Derivatives, differentials and Taylor Theorem</li> <li>Riemann integral</li> <li>Logarithmic and exponential functions</li> <li>Limits of functions and continuity</li> </ul>
Artificial Intelligence (cohort till 2024)
Module 1: Foundation of Artificial Intelligence <ul style="list-style-type: none"> <li>Searching for Solutions</li> <li>Reasoning with Constraints</li> <li>Agents based on Propositional Logic</li> <li>Planning with Certainty</li> <li>Multiagent Systems and Games</li> <li>Large Language Models as Agents</li> </ul>
Module 2: Machine Learning in Practice <ul style="list-style-type: none"> <li>Data understanding and preprocessing</li> <li>Classification: Decision Trees, Rule-based classification, KNN, Naïve Bayes, Support Vector Machines, Perceptron</li> <li>Ensemble learning, boosting, bagging (Random Forests)</li> <li>Evaluation of Machine Learning algorithms</li> <li>Regression analysis</li> <li>K-Means Clustering</li> </ul>
Artificial Intelligence (cohort from 2025 – course starting from a.y. 26/27)
Module 1: Foundation of Artificial Intelligence <ul style="list-style-type: none"> <li>Artificial Intelligence and Agents</li> <li>Searching for Solutions</li> <li>Reasoning with Constraints</li> <li>Propositions and inference</li> <li>Planning with Certainty</li> </ul>

<ul style="list-style-type: none"> <li>• Multiagent Systems and Games</li> </ul>
<p>Module 2: Introduction to Computer Vision</p> <ul style="list-style-type: none"> <li>• Fundamentals of image formation</li> <li>• Preprocessing, feature detection and matching</li> <li>• Image registration, stereo and multi-view reconstruction</li> <li>• Image classification, object detection, segmentation</li> <li>• Motion estimation and tracking, action recognition</li> <li>• Advanced topics and applications</li> </ul>
<p>Computational Mathematics (cohort from 2024 - course starting from a.y. 26/27) Computational Mathematics for Data Science (cohort from 2025 – course starting from a.y. 27/28)</p> <ul style="list-style-type: none"> <li>• Introduction to Computational Modelling and Finite Precision Computation</li> <li>• Matrix Factorization Methods: LU, Cholesky, and QR Factorization</li> <li>• Data-Driven Approaches in Data Science: Principal Component Analysis (PCA), Data Compression (SVD), and Linear and Nonlinear Regression</li> <li>• Iterative Methods for Solving Nonlinear Equations and Optimization</li> <li>• Numerical Methods for Solving Ordinary Differential Equations (ODEs)</li> <li>• The Google PageRank Problem and Numerical Methods for Eigenvalue Computation</li> </ul>
<p>Computational Security (cohort till 2023)</p>
<p>Module 1: Computational Mathematics</p> <ul style="list-style-type: none"> <li>• Introduction to computational modeling and principles of finite precision computation</li> <li>• Matrix factorization methods: LU, Cholesky, QR, and singular value decomposition</li> <li>• Applications in data science: data compression, principal component analysis, regression</li> <li>• Iterative methods for solving nonlinear equations and optimization problems</li> <li>• Numerical methods for solving ordinary differential equations</li> <li>• The Google PageRank problem and its applications: power method, shifted power method, and inverse power method</li> </ul>
<p>Module 2: Information Security</p> <ul style="list-style-type: none"> <li>• Basic definitions: CIA, threat, attack, vulnerability, access control</li> <li>• Risk assessment</li> <li>• Basics of cryptography</li> <li>• Network attack and defense</li> <li>• Usability</li> <li>• Security policies</li> </ul>
<p>Computer Networks</p> <ul style="list-style-type: none"> <li>• Introduction to computer networks</li> <li>• ISO OSI reference model</li> <li>• Internet applications and application protocols (HTTP, SMTP, DNS)</li> <li>• Network protocols: TCP/IP, Ethernet</li> <li>• Sockets and RPCs</li> <li>• Failure robustness, security</li> </ul>

<p>Computer Programming cohort till 2024 - Introduction to Programming (cohort from 2025)</p> <ul style="list-style-type: none"> <li>• Data types and expressions</li> <li>• Basic data structures and generic</li> <li>• Functions and parameter passing</li> <li>• Conditionals and loops</li> <li>• Arrays and collections</li> <li>• Classes and objects</li> <li>• Basic Input/Output</li> <li>• Exception handling</li> <li>• Recursion</li> </ul>
<p>Computer Systems Architecture</p> <ul style="list-style-type: none"> <li>• Computer abstractions and technology</li> <li>• Bits, datatypes, and arithmetic in computer systems</li> <li>• Gates, circuits, and combinational logic</li> <li>• Sequential logic</li> <li>• A simple processor architecture: the CPU</li> <li>• Instruction sets and assembly language</li> </ul>
<p>Database Management Systems (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Physical data storage</li> <li>• Indexing and hashing</li> <li>• Query processing and optimization</li> <li>• Transaction processing</li> <li>• Concurrency control</li> <li>• Recovery</li> </ul>
<p>Databases (cohort from 2025 – course starting from a.y 26/27)</p>
<p>Module 1: Introduction to Databases</p> <ul style="list-style-type: none"> <li>• Relational data model and relational algebra</li> <li>• The SQL language</li> <li>• Using SQL in database applications: API, embedded SQL</li> <li>• The Entity Relationship model</li> <li>• Conceptual database design</li> <li>• Logical database design</li> </ul>
<p>Module 2: Database Management Systems</p> <ul style="list-style-type: none"> <li>• Physical data storage</li> <li>• Indexing and hashing</li> <li>• Query processing and optimization</li> <li>• Transaction processing</li> <li>• Concurrency control</li> <li>• Recovery</li> </ul>
<p>Data Structure and Algorithms</p> <ul style="list-style-type: none"> <li>• Design Principles: Problem reduction via recursion</li> </ul>

<ul style="list-style-type: none"> <li>• Searching and Sorting</li> <li>• Correctness: Loop invariants, termination</li> <li>• Complexity: Asymptotic analysis</li> <li>• Divide and Conquer</li> <li>• Pointers, dynamic data structures, linked lists</li> <li>• Abstract data types: stacks, queues, priority queues, maps</li> <li>• Binary trees, red-black trees</li> <li>• Graph algorithms</li> </ul>
<p>Discrete Mathematics</p> <ul style="list-style-type: none"> <li>• Elements of logic, propositions and quantifiers, methods of mathematical proof</li> <li>• Numbers and basic number theory</li> <li>• Set Theory, Russell Paradox and Halting Problem</li> <li>• Functions, infinite cardinalities and countability</li> <li>• Relations, orders, equivalence classes</li> <li>• Graphs and trees</li> </ul>
<p>English for Computer Scientists 1 (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• General overview of grammatical structures at the C1 level;</li> <li>• Development of receptive skills through the exposure to and analysis of various types of written and spoken discourse typical in Computer Science and development of grammatical and lexical range and accuracy so that communication is fluent and spontaneous;</li> <li>• Vocabulary acquisition and word-building techniques; lexicogrammar.</li> </ul>
<p>English for Computer Scientists 2 (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Study skills: focus on developing the specific study skills that are required in this undergraduate programme in Computer Science including critical thinking skills;</li> <li>• Writing skills: practice of coherent academic discourse to produce subject-specific texts in English at the C1 level, including formal academic emails, reports and summaries;</li> <li>• Spoken skills: improvement of spoken interaction and production through the practice and production of academically and professionally acceptable presentations and other domain-specific speaking activities.</li> </ul>
<p>English for Computer Scientists (cohort from 2025)</p> <ul style="list-style-type: none"> <li>• Specialised grammar, syntax and lexis at C1 level: complex sentences; lexicogrammar;</li> <li>• Development of productive skills through the exposure to and analysis of various types of written and spoken discourse typical in Computer Science and development of grammatical and lexical range and accuracy so that written and spoken communication is fluent and spontaneous;</li> <li>• Audience and effects on language register and style.</li> </ul>
<p>Formal Languages and Compilers</p> <ul style="list-style-type: none"> <li>• Formal language theory</li> <li>• Regular languages: automata, regular expressions, regular grammars</li> <li>• Context free languages (stack machines)</li> <li>• Lexical and syntactic analysis: Lexer specification, top-down and bottom-up parsing</li> <li>• Semantic analysis: Rules for type checking, symbol table and control flow</li> </ul>

<ul style="list-style-type: none"> <li>Intermediate code generation</li> </ul>
<p>German for Computer Scientists (cohort till 2024)</p> <ul style="list-style-type: none"> <li>Listening skills: comprehension of talks, documentary, reportings, descriptions in different contexts, on different media, about ICT topics</li> <li>Writing skills: practice of coherent academic discourse to produce subject-specific texts (for example application letter, report, product review, compliant mail, instructions, essay, abstract, summary, seminar work etc.) about ICT topics;</li> <li>Spoken skills: improvement of spoken interaction and production through the practice and production of academically and professionally acceptable presentations and other domain-specific speaking activities;</li> <li>Development of receptive skills (reading and listening, both global and detailed) through the exposure to and analysis of various types of authentic written and spoken discourse typical in Computer Science and development of grammatical and lexical range and accuracy so that communication is fluent and spontaneous;</li> <li>Language mediation (mediating communication, text and concepts) from English to German and viceversa about area of expertise (ICT);</li> <li>Vocabulary acquisition and word-building techniques; lexicogrammar.</li> </ul>
<p>German for Computer Scientists (cohort from 2025 – course active from a.y. 27/28)</p> <ul style="list-style-type: none"> <li>Writing skills: practice of coherent academic discourse to produce subject-specific texts; practice of all communication texts, such as e-mails, web texts;</li> <li>Spoken skills: improvement of spoken interaction and production through the practice and production of academically and professionally acceptable presentations and other domain-specific speaking activities;</li> <li>Development of receptive skills through the exposure to and analysis of various types of written and spoken discourse typical in ICT and development of grammatical and lexical range and accuracy so that communication is fluent and spontaneous;</li> </ul>
<p>Information Security (cohort from 2025 – offered a.y. 27/28)</p> <ul style="list-style-type: none"> <li>Basic definitions: CIA, threat, attack, vulnerability, access control</li> <li>Risk assessment</li> <li>Basics of cryptography</li> <li>Network attack and defense</li> <li>Usability</li> <li>Security policies</li> </ul>
<p>Intelligent Agents</p>
<p>Module 1: Knowledge Representation</p> <ul style="list-style-type: none"> <li>Propositional and First-Order Languages</li> <li>Individuals and Relations</li> <li>Knowledge Representation and Logic</li> <li>Knowledge-Based Systems: Description Logics and Ontologies</li> <li>Non-classical Logic and Formal Reasoning</li> <li>Common-Sense Knowledge</li> </ul>

## Module 2: Intelligent Agents Project

- AI paradigms: symbolic approaches vs. learning-based approaches
- Overview of main AI techniques: exact and approximate methods, handling imperfect information, use and model of domain knowledge
- Tools and programming techniques for the development of AI systems
- Symbolic approaches: planning and search, constraint solving, description logic and ontologies, multi-agent models
- Learning-based approaches: supervised vs. unsupervised and reinforcement learning, neural networks
- Hands-on programming projects covering the above topics

## Interactive Interface Design and Development (cohort from 2023)

- Fundamentals of interaction design.
- Fundamental of interaction design principles for web apps.
- Fundamental of interaction design patterns for web apps.
- Interaction design prototyping techniques for web apps.
- Interaction design prototyping tools for web apps.
- Fundamentals of web programming, client-side, for web apps

## Introduction to Business Administration (cohort from 2020 to 2022)

- Introduction to management: company overview and business functions, capital configuration, company equilibrium, organizational dynamics
- Accrual Accounting and Financial Statements
- Managerial accounting for decision making (costing and pricing), planning, budgeting and reporting

## Introduction to Databases (cohort till 2024)

- Relational data model and relational algebra
- The SQL language
- Using SQL in database applications: API, embedded SQL
- The Entity Relationship model
- Conceptual database design
- Logical database design

## Italian for Computer Scientists (cohort till 2024)

- Listening skills: comprehension of talks in different contexts, live, by phone or other media, about ICT topics
- Writing skills: practice of coherent academic discourse to produce subject-specific texts; practice of all communication texts, such as e-mails, web texts;
- Spoken skills: improvement of spoken interaction and production through the practice and production of academically and professionally acceptable presentations and other domain-specific speaking activities;
- Development of receptive skills through the exposure to and analysis of various types of written and spoken discourse typical in ICT and development of grammatical and lexical range and accuracy so that communication is fluent and spontaneous;
- Language mediation (mediating communication, text and concepts) from English to German and viceversa about area of expertise (ICT);
- Vocabulary acquisition and word-building techniques; lexicogrammar.

<p>Italian for Computer Scientists (cohort from 2025 – course active from a.y. 27/28)</p> <ul style="list-style-type: none"> <li>• Writing skills: practice of coherent academic discourse to produce subject-specific texts; practice of all communication texts, such as e-mails, web texts;</li> <li>• Spoken skills: improvement of spoken interaction and production through the practice and production of academically and professionally acceptable presentations and other domain-specific speaking activities;</li> <li>• Development of receptive skills through the exposure to and analysis of various types of written and spoken discourse typical in ICT and development of grammatical and lexical range and accuracy so that communication is fluent and spontaneous;</li> </ul>
<p>Linear Algebra</p> <ul style="list-style-type: none"> <li>• Background on complex numbers, trigonometry and polynomials</li> <li>• Vectors and matrices</li> <li>• Linear Systems</li> <li>• Vector spaces</li> <li>• Linear operators</li> <li>• Spectral analysis</li> </ul>
<p>Maker Lab (cohort from 2020 to 2022)</p> <ul style="list-style-type: none"> <li>• Basics of programming for physical computing and interactions with the world using Python 3</li> <li>• Basics of electronics for physical computing: interruptors, sensors, actuators and hats</li> <li>• Basics of interaction design and development for physical computing</li> </ul>
<p>Machine Learning (cohort from 2025/26)</p> <ul style="list-style-type: none"> <li>• Data understanding and preprocessing</li> <li>• Classification: Decision Trees, Rule-based classification, KNN, Naïve Bayes, Support Vector Machines, Perceptron</li> <li>• Ensemble learning, boosting, bagging (Random Forests)</li> <li>• Evaluation of Machine Learning algorithms</li> <li>• Regression analysis</li> <li>• K-Means Clustering</li> </ul>
<p>Mobile and Physical Systems</p>
<p>Module 1: Engineering of Mobile Systems (cohort till 2024) Module 1: Mobile Computing (cohort from 2025 – course starting from a.y. 27/28)</p> <ul style="list-style-type: none"> <li>• Functional and declarative programming</li> <li>• Design of mobile applications</li> <li>• Frameworks and platforms for mobile development</li> <li>• Data and resource management in a mobile context</li> <li>• Mobile device sensors</li> <li>• Internet of Things</li> </ul>
<p>Module 2: Physical Computing Project (cohort from 2020 to 2022) / Prototyping Physical Interactive Experiences (cohort from 2023 – starting from a.j. 25/26)</p> <ul style="list-style-type: none"> <li>• Motivation &amp; Short overview about prototyping (using ProtoPie)</li> <li>• Sensor Fundamentals (using Arduino/ESP32)</li> </ul>

<ul style="list-style-type: none"> <li>• 3D Environment Development (using Unity)</li> <li>• Design &amp; sketch a next-generation controller</li> <li>• Presentation of the prototype</li> <li>• Evaluation of the prototype</li> </ul>
Network Security (cohort 2024- starting from a.j. 26/27)
<p>Module 1: Computer Networks</p> <ul style="list-style-type: none"> <li>• Introduction to computer networks</li> <li>• ISO OSI reference model</li> <li>• Internet applications and application protocols (HTTP, SMTP, DNS)</li> <li>• Network protocols: TCP/IP, Ethernet</li> <li>• Sockets and RPCs</li> <li>• Failure robustness, security</li> </ul>
<p>Module 2: Information Security</p> <ul style="list-style-type: none"> <li>• Basic definitions: CIA, threat, attack, vulnerability, access control</li> <li>• Risk assessment</li> <li>• Basics of cryptography</li> <li>• Network attack and defense</li> <li>• Usability</li> <li>• Security policies</li> </ul>
<p>Operating Systems (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Programming in C</li> <li>• Scheduling and concurrency</li> <li>• Processes and synchronization</li> <li>• File systems and memory management</li> <li>• Storage management</li> <li>• Security and protection</li> </ul>
Operating Systems (cohort from 2025)
<p>Module 1: Operating Systems and Networking</p> <ul style="list-style-type: none"> <li>• Operating Systems Structures</li> <li>• Processes, Threads and Concurrency</li> <li>• CPU Scheduling and Synchronization</li> <li>• Memory and Mass-Storage</li> <li>• I/O, File Systems</li> <li>• Networks and Distributed Systems</li> </ul>
<p>Module 2: Programming in C</p> <ul style="list-style-type: none"> <li>• Data types, variables, operators, control structures (loops, conditionals), functions, and pointers</li> <li>• Dynamic memory allocation, arrays, structures, linked lists, stacks, and queues</li> <li>• File handling, bitwise operations, multi-file programs, debugging, and optimization</li> </ul>
<p>Probability Theory and Statistics</p> <ul style="list-style-type: none"> <li>• Basic concepts: probability spaces, conditional probability, Bayes' Theorem, independent events</li> </ul>



<ul style="list-style-type: none"> <li>• Random variables: distribution, density, expectation, variance, covariance, law of large numbers</li> <li>• Special distributions: Bernoulli, Binomial, Poisson, Exponential, Normal, Chi-Square, t-Distribution</li> <li>• Sampling: sums of random variables, central limit theorem, sample variance</li> <li>• Parameter Estimation: maximum likelihood estimates, interval estimates, confidence intervals</li> <li>• Hypothesis testing: significance levels, test statistics, p-values</li> </ul>
<p>Programming Project (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Objects and Classes: Interfaces, Inheritance and Polymorphism</li> <li>• Abstract data types (set, list, map, queue)</li> <li>• Dynamic Data structures (linked list, hashmap)</li> <li>• Recursion</li> <li>• Serialisation and streams</li> <li>• Mutability, pure functions and lambda expressions</li> <li>• Multithreading</li> <li>• IDE: Git, Source code management, and build automation</li> <li>• Project</li> </ul>
<p>Object Oriented and Functional Programming (cohort from 2025)</p> <ul style="list-style-type: none"> <li>• Object-oriented design: objects, classes, interfaces, inheritance and polymorphism</li> <li>• Abstract data types (set, list, associative array, queue, ...) and related data structures (linked list, hash table, ...),</li> <li>• Composite types, type inference, generics</li> <li>• Basic input/output, serialization, streams, error handling, custom exceptions, debugging</li> <li>• Introduction to functional programming: function composition, recursion, currying, closures, function types</li> <li>• Functional principles applied to object-oriented programming: immutability, pure functions, lambda abstractions</li> </ul>
<p>Project and Teamwork Management</p> <ul style="list-style-type: none"> <li>• Project and team work management methods and techniques: goal specification techniques, coordination and collaboration techniques, performance and risk management</li> <li>• Human resources management: communication, conflict management</li> <li>• Tool support for project and team work management</li> </ul>
<p>Scientific Writing and Communication (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Planning communication: audiences, media and presenters</li> <li>• In-reach communication: structure of thesis, journal paper, posters, presentation</li> <li>• Out-reach communication: technical manuals, writing for the web; press release; social media; public dissemination</li> </ul>
<p>Scientific Writing and Communication in English (cohort from 2025 – course starting from a.y. 27/28)</p> <ul style="list-style-type: none"> <li>• Study skills: focus on developing the specific study skills that are required in this undergraduate programme in Computer Science including critical thinking skills and specific reading skills;</li> <li>• Writing skills: practice of coherent academic and scientific discourse to produce subject-specific texts in English at the C1 level, including academic reports and summaries;</li> <li>• Spoken skills: improvement of spoken interaction and production through the practice and production</li> </ul>

of academically and professionally acceptable presentations and other domain-specific discussions.
<p>Software Engineering (cohort till 2024)</p> <ul style="list-style-type: none"> <li>• Software life-cycle: principles and methodologies</li> <li>• Software processes and software project management</li> <li>• Requirements engineering: elicitation and modeling</li> <li>• System modeling and construction: UML, design patterns</li> <li>• Software testing and management: principles and techniques</li> <li>• Recent software engineering topics: DevOps, Cloud, SE and AI</li> </ul>
Software Engineering with Project (cohort from 2025, course starting from a.y. 26-27)
<p>Module 1: Software Engineering</p> <ul style="list-style-type: none"> <li>• Software life-cycle: principles and methodologies</li> <li>• Software processes and software project management</li> <li>• Requirements engineering: elicitation and modeling</li> <li>• System modeling and construction: UML, design patterns</li> <li>• Software testing and management: principles and techniques</li> <li>• Recent software engineering topics: DevOps, Cloud, SE and AI</li> </ul>
<p>Module 2: Coding Capstone</p> <ul style="list-style-type: none"> <li>• Project Inception &amp; Roadmap</li> <li>• Architecture &amp; Technical Design</li> <li>• Development &amp; Integration</li> <li>• Deployment &amp; Observability</li> <li>• Final Review &amp; Retrospective</li> <li>• Cross-Team Collaboration &amp; Stakeholder Engagement</li> </ul>
Software Architecture (cohort from 2020 to 2022) Software Systems Engineering (cohort from 2023 to 2024, starting a.y 24/25)
<p>Module 1: Software Systems Architecture</p> <ul style="list-style-type: none"> <li>• Software and systems architecture principles</li> <li>• Software architecture design process</li> <li>• Architectural components and frameworks</li> <li>• Approaches for architectural partitioning</li> <li>• Architectural patterns and styles</li> <li>• Integrating AI Components into Architectural Designs</li> </ul>
<p>Module 2: Tools and Techniques for Software Testing</p> <ul style="list-style-type: none"> <li>• Verification and validation</li> <li>• Techniques for black-box and white-box testing</li> <li>• Techniques and tools for test automation</li> <li>• Integration and regression testing</li> <li>• Web testing</li> <li>• AI for test case generation</li> </ul>
Smart Software Systems (cohort from 2025 course active from a.y. 26-27)

#### Module 1: Smart Software Design

- Design Patterns Application and Interaction
- Refactoring
- Design Assessment
- Frameworks, Reflection and Metadata
- Multithreading software design
- AI-assisted application design

#### Module 2: Smart Software Development

- Design Patterns Application and Interaction
- Refactoring
- Design Assessment
- Frameworks, Reflection and Metadata
- Multithreading software design
- AI-assisted application design

#### Web and Internet Engineering (cohort till 2024) Web and Internet Engineering with Project (cohort from 2025)

- Basics of computer networks, Web protocols and markup languages
- Development of web applications: basics of usability, accessibility and responsive design
- Client-side dynamicity and web scripting languages
- Client-side GUI frameworks
- Web application design and web services
- Languages and frameworks for server-side web development