

Faculty of Mathematics and Information Science
Warsaw University of Technology

Data Science

Undergraduate Studies (Bachelor of Science – B.Sc.)

Seven-Semester Full-Time Programme

General Course Syllabi

Academic years: 2022–2026

Duration: 7 semesters (3.5 years)

Total ECTS points: 210

Mode of study: Full-time

Language of instruction: Polish

Contents

Semester 1	3
Calculus 1	3
Computer Architecture	3
Creative Problem Solving	4
Introduction to Formal Logic and Set Theory	4
Introduction to Programming and Data Processing	4
Linear Algebra with Geometry	4
Operating Systems in Data Engineering	5
Semester 2	5
Algebra in Data Science	5
Calculus 2	5
Discrete Mathematics and Elements of Probability	6
Object-Oriented Programming	6
Selected Issues in the Philosophy of Information and Computer Science	7
Structured Data Processing	7
Language Course 1 & 2	7
Physical Education and Sports 1	7
Semester 3	7
Advanced Object-Oriented and Functional Programming	7
Algorithms and Data Structures 1	7
Data Visualisation Techniques	7
Numerical Methods	8
Physics 1	8
Probability	9
Language Course 3 & 4	9
Physical Education and Sports 2	9
Semester 4	9
Research Workshop 1	9
Databases	9
Introduction to Machine Learning	10
Mathematical Statistics	10
Physics 2	10
Elective 1	11
Language Course 5 & 6	11
Physical Education and Sports 3	11
Semester 5	11
IT Systems Engineering	11
Methods of Computational Statistics	11
Optimisation Methods	11
Stochastic Processes	12
Web Application Programming	12
Computer Networks (elective block)	12
Elective 2	12

Semester 6	12
Research Workshop 2	12
Computational Intelligence Methods in Data Analysis	13
Data Warehouses and Business Intelligence	13
Interdisciplinary Project	13
IT Systems Architecture	13
Elective 3	13
Elective 4	13
Semester 7	13
Data Storage in Big Data Systems	13
Data Transmission	14
Diploma Seminar	14
Group Project	14
Introduction to Financial Management	14
Diploma Thesis	14

Semester 1

Calculus 1

Bounded sets and their bounds (supremum and infimum). Real sequences. Definition of the limit of a sequence. Monotonic sequences and theorems on their convergence. Bounded sequences and the Bolzano-Weierstrass theorem. Limits of a sequence. Comparison of sequences. Indeterminate forms. Sequences divergent to infinity. Limit superior and limit inferior.

Real-valued functions of a single real variable. Boundedness, monotonicity, and bijectivity of functions. Composite functions and inverse functions; the relationship between their graphs. Heine and Cauchy definitions of the limit of a function. Improper limits, limit theorems, the theorem on the preservation of inequalities in the limit, and the squeeze theorem. Continuous functions and theorems on continuous functions. One-sided limits and one-sided continuity. Relationships with limits. Vertical, horizontal, and oblique asymptotes.

Polynomials and radical functions. Trigonometric functions and their inverses (cyclic functions). Reduction formulas and trigonometric identities. Exponential and logarithmic functions; the exponential function and its inverse, the natural logarithm. Hyperbolic functions and their inverses.

The sign-preserving theorem for continuous functions. The Darboux property (Intermediate Value Theorem). Weierstrass theorem on the existence of extreme values for continuous functions. Uniform continuity. Cantor's theorem.

Definition of the derivative of a function and differentiable functions. One-sided derivatives. Geometric interpretation of the derivative. Theorems on the derivative of the sum, product, and quotient of two functions. Chain rule for the derivative of a composite function. Inverse function theorem for derivatives. Derivation of formulas for the derivatives of elementary and inverse functions. Higher-order derivatives and differentials. Rolle's theorem. Cauchy's mean value theorem. Lagrange's mean value theorem and its applications to function monotonicity. Taylor's theorem (Maclaurin formula). Function approximation by polynomials and the approximation error. Calculating limits using de l'Hospital's rule. Extrema of a function and the necessary condition for their existence. Two theorems on the sufficient conditions for the existence of extrema. Definition of convex and concave functions. Relationship between function convexity and the second derivative. Inflection points and the necessary condition for their existence. Function analysis and its graph.

Definition of the antiderivative and the indefinite integral. Theorems on integrable functions. Integration by substitution. Integration by parts. Recursive integration formulas. Integration of rational functions. Integration of trigonometric and irrational functions.

Computer Architecture

Binary logic and numerical codes. Data representation: integers and floating-point numbers. Fundamentals of digital arithmetic. Overview of computer architectures. Concepts of system and hardware mechanisms. Organisation of: buses, arbitration, DMA, instruction decoding and sequencer operation, ALU.

Processor systems. CISC and RISC architectures. SISD, SIMD, and MIMD processing. Processor architectures. Pipeline processing. Superscalar architecture.

Memory, cache, and memory hierarchy. I/O space, interrupts, and communication with external devices. MMU (Memory Management Unit). Process memory protection. Virtualisation. Microprocessor architectures and examples. Program memory model, compilation, execution stack, index registers, and heap.

Structure and operation of a microkernel, system stack, context switching, and multiprocessor microkernel.

Creative Problem Solving

Design Thinking and Problem-Based Learning methodologies. Work organisation and team collaboration. Identifying user needs. Problem definition. Generating solutions through brainstorming. Prototyping and testing.

Introduction to Formal Logic and Set Theory

Introduction to logic, propositional calculus, and predicate calculus. Set theory. Indexed families of sets. Union and intersection of a family of sets. Relations, especially equivalence relations, equivalence classes, order relations, Hasse diagrams, bounds. Functions, image, preimage. Construction of natural, integer, and rational numbers. Cardinality of sets, countable sets and their properties.

Introduction to Programming and Data Processing

The concept of a problem and an algorithm. Scalar types: bool, int, float, str. Representation of integers and floating-point numbers. Logical, arithmetic, relational, and assignment operators. Operator precedence. Floating-point arithmetic errors. Conditional statements and loops. Measures of algorithmic complexity. Program structure. Functions. Code documentation and testing. Passing parameters by value vs. by reference.

Implementation of algorithms: bisection method for finding function roots, greatest common divisor, primality testing, etc. Lists (sequences, arrays). Implementation of algorithms: applying operations to list elements, calculating cumulative sums and iterated differences, Cartesian product operations on two lists, operations on corresponding elements of multiple lists; simple sorting methods, bucket sort, finding the mode in a list, counting unique elements in a vector, etc. Recursion. Backtracking algorithms. Divide and conquer algorithms. Implementation of algorithms: Towers of Hanoi, 8 queens problem, binary search, finding the closest pair of points in R^2 , etc.

Elements of object-oriented programming: simple classes, fields, and methods. Mutable and immutable types. Iterable and sequential types. Dictionaries and sets. Implementation of algorithms: element counting, calculating means within subgroups. Vectors and matrices (NumPy). Vectorised operations on vectors and matrices (tensors). Implementation of simple algorithms using library functions and methods based on documentation, e.g., 1-nearest neighbours, simple simulation experiments, etc.

Linear Algebra with Geometry

Algebraic systems: groups (permutation groups), rings (Z_n), fields; the field of complex numbers. Systems of linear equations, matrices, elementary row (column) operations on matrices; elementary matrices. Gaussian elimination method. Matrices, matrix operations, matrix equations $AX = B$. Linear spaces. Subspaces. linear dependence and independence of vectors, basis, dimension of a linear space. Matrix rank. The Kronecker-Capelli (Rouché–Fontené–Frobenius) theorem. Homomorphisms of linear spaces. Kernel, image.

Matrices of homomorphisms. Isomorphisms and invertible matrices. Determinants. Applications of determinants. Matrix factorisations. Eigenvalues and eigenvectors of matrices and linear operators. Characteristic polynomial. Diagonalisation of matrices and linear operators. Bilinear and Hermitian forms. Positive and negative definiteness of bilinear forms. Matrices of forms.

Operating Systems in Data Engineering

Upon completion of the course, students should: have general knowledge of the development branches of the Unix system (System V and BSD), as well as POSIX and SUS standards; have basic knowledge of FLOSS software, free licenses, and *BSD and GNU/Linux systems; be able to use the shell (bash); know basic Unix commands; be able to use the Unix system help (man); have basic knowledge of operating system architecture (Unix), with a particular emphasis on file organisation and process management; have elementary knowledge of Unix system configuration and administration principles; be able to manage services (daemons); be able to use CRON and SYSLOG services; use the vim editor at a basic level; describe string patterns using POSIX ERE regular expressions; be able to write simple scripts in bash and AWK; be able to download from remote sources, filter, merge, and split files.

Using and customising graphical environments based on the GTK+ library; users, groups, identifiers, and permissions; basic commands; using the shell (bash); file systems, file types, directory structure, access permissions, ACLs; fundamentals of process management; system configuration and selected configuration files; standard services (CRON, SYSLOG, ...); working in the vim editor; basics of bash programming; regular expressions and text processing with the SED editor; programming in AWK; data retrieval and processing tools such as curl, head, tail, grep, cut, and CSV file processing.

Semester 2

Algebra in Data Science

Matrices and matrix operations. Information encoding—linear codes. Orthogonality in linear spaces, orthogonal projections, and orthogonal matrices. Eigenvalue problems and their applications. Matrix factorisations: QR, LU, diagonalisation, and Singular Value Decomposition (SVD). (Semi-)positive/negative definiteness of matrices and the application of such matrices. Matrix norms.

Calculus 2

Integral sums and the definition of the Riemann integral. Upper and lower Darboux integrals. Theorems on Riemann-integrable functions. Properties of the Riemann integral. Geometric interpretation of the Riemann integral. Definition of the definite integral and its properties. The fundamental theorem of calculus. The Newton-Leibniz formula. Integration by substitution. Integration by parts. Mean value theorem for integrals. Recurrence formulas for certain definite integrals. Definition of improper integrals of the first kind; Cauchy principal value. Definition of improper integrals of the second kind. Calculation of areas of normal regions. Calculation of the arc length of a rectifiable curve. Calculation of surface areas and volumes

of solids of revolution.

Definition of a series, its partial sums, and the sum of a series. Dirichlet series and geometric series. Necessary condition for series convergence. Comparison test for convergence of series with non-negative terms. Series with arbitrary terms; absolute convergence. Cauchy test and D'Alembert test for series convergence. Alternating series, Leibniz criterion, and conditional convergence. Rearrangement of terms in series. Riemann series theorem.

Pointwise and uniform convergence of function sequences. Properties of uniformly convergent sequences. Pointwise and uniformly convergent a series of functions. Weierstrass M-test. Term by term integration and differentiation of a series of functions. The Abel theorem. Radius of convergence of power series; The Cauchy–Hadamard theorem. Taylor and Maclaurin series. Taylor series expansion. Maclaurin series expansion of exponential, sine, and cosine functions. Euler's formulas. Trigonometric Fourier series. Dirichlet conditions. Dirichlet's theorem. Expansion of functions into sine and cosine series.

Definition of a metric. Examples of various metrics. Definitions of a ball and the distance from a point to a set. Convergence in metric spaces. Completeness. Definition of a norm and its properties. Banach fixed-point theorem. Method of successive approximations. Limits and continuity of functions of several variables. Differentiable functions, directional derivatives, partial derivatives, and the gradient of a function. Application of differentials for approximate calculations. Higher-order derivatives and differentials. The Schwarz theorem. Differentiation of composite functions of one and several variables. Taylor's formula for functions of several variables. Necessary condition for the existence of an extremum. Sufficient conditions for the existence of an extremum of functions of several variables. Absolute maximum and minimum values of functions of several variables. Definition of implicit functions of several variables. Implicit function theorem. Theorem on the derivative of an implicit function. Necessary and sufficient conditions for the existence of an extremum of an implicit function of one variable.

Vector-valued functions of one variable, continuity, and differentiability. Geometric interpretation of the derivative of a vector function. Vector-valued functions of several variables, continuity, and differentiability. Graphs of vector functions—description of surfaces. The Jacobian matrix. The Jacobian of a transformation. Polar, cylindrical, and spherical coordinates. Regular surface patches and the tangent plane to a surface given in explicit and parametric forms. Definition and properties of differential operators: gradient, divergence, and curl. Jordan measure in R^n . Definition and properties of the Riemann integral. Normal regions. Iterated integrals. Change of variables theorem (integration by substitution). Double and triple integrals. Geometric interpretation of double and triple integrals.

Discrete Mathematics and Elements of Probability

Fundamentals of combinatorics: counting methods, inclusion-exclusion principle, generating functions, recursion. Fundamentals of probability: discrete probability space, conditional probability, independence of events, random variables, expected value, variance, binomial distribution. Fundamentals of graph theory: trees, minimum spanning tree, graph connectivity, Eulerian and Hamiltonian cycles, graph colouring, planar graphs.

Object-Oriented Programming

Introduction. Simple types, strings, arrays. Classes and objects. Features of object-oriented programming. Interfaces. Enumeration types. Exception handling. Elements of generic programming. Collections. Input/output mechanisms. Graphics in Java. Event handling. User interface. Threads. Code documentation. Preparation and distribution of applications.

Selected Issues in the Philosophy of Information and Computer Science

Philosophical problems related to computer science, the theory of computation, the concept of information, artificial intelligence, etc., including the ethical aspects generated by the development of information technologies.

Structured Data Processing

Basic atomic data types in R: Vectors and NULL. Vectorised operations on atomic vectors. Transforming and filtering variables. Aggregation and graphical presentation of variables. Text data processing. Regular expressions. Lists. Functions. Control flow statements and loops. Object attributes. S3-style object-oriented programming. Complex types: objects representing time, factors, time series, matrices, and data frames, and basic operations on them. Non-standard evaluation. Formulas. Filtering, transforming, and cleaning data frames. Data imputation. Aggregation and other operations on data in subgroups. Merging data frames. File and directory operations. Fetching data from APIs. Web scraping. Data wrangling in Python with: numpy, scipy, pandas, matplotlib, and seaborn.

Language Course 1 & 2

Physical Education and Sports 1

Semester 3

Advanced Object-Oriented and Functional Programming

Time representation. Inner and anonymous classes. Lambda expressions and functional interfaces. Data processing. Network applications. Reflection and annotations. Localisation. Design patterns.

Algorithms and Data Structures 1

Analysis of computational complexity: The concept of computational complexity. Asymptotic notation. Common scenarios (worst-case, best-case, average, amortised). Orders of complexity magnitude, example algorithms. Problem classes P, NP, NPC. Algorithm design methods: Divide and conquer strategy, recursion, dynamic programming, greedy algorithms, backtracking, heuristic algorithms.

Priority queues. Heap and d-ary heap. Mergeable heaps. Binomial queues, Fibonacci heaps.

Dictionaries. Array searching. Search trees: BST, AVL, red-black trees, optimal trees, self-organising trees. B-trees, 2-3 and 2-3-4 trees. Radix search, Hashing.

Union-find algorithms. List representation. Tree representation.

Sorting. Internal sorting by comparison. Radix sort. Counting sort. External sorting. Selection problem: Hoare's algorithm (Quickselect), Median-of-Medians algorithm.

Data Visualisation Techniques

History of static graphics. Colour perception. Selection of chart element features (length, area, angles, colours) for variables measured according to different scales (ratio, interval, ordinal,

nominal). Software for preparing static graphics, specifically the `ggplot2` package in R and the `matplotlib` library in Python. Software for preparing interactive graphics using the `plotly` package in R and Python. Software for preparing interactive applications, specifically the `shiny` package for R. Examples of successful and unsuccessful static graphics from media and scientific articles.

Libraries for creating static charts: `ggplot2`, `ggstatsplot`, `ggrepel`, `maps`, `patchwork`, `matplotlib`, `seaborn`. Libraries for creating interactive graphics: `gganimate`, `ggiraph`, `visNetwork`, `plotly`. Libraries for creating interactive applications: `shiny`.

Numerical Methods

Errors in numerical calculations. Absolute and relative error. Conditioning of a numerical problem. Error of function values. Arithmetic operation errors: addition and subtraction, multiplication, division. Machine representation of floating-point numbers. Properties of basic arithmetic operations. Error calculus. Numerical correctness of algorithms.

Calculating the value of a polynomial and its derivatives. Horner's method. Calculating the first derivative of a polynomial. Calculating the second derivative of a polynomial. Finding roots of single-variable functions. Newton's method (tangent method). Secant method. Halley's method. Parabolic method. Bisection method. Convergence rate of iterative methods. Convergence exponent values.

Interpolation. Polynomial interpolation. Lagrange form of the interpolating polynomial. Newton form of the interpolating polynomial, divided difference scheme. Generalised Horner scheme. Interpolation error, convergence of interpolation processes, node selection. Hermite interpolation.

Numerical integration. Quadratures. Interpolatory quadratures. Newton-Cotes quadratures (simple): trapezoidal, Simpson's, Newton 3/8, rectangular. Composite quadratures: trapezoidal, Simpson's, rectangular, Newton. Change in integration interval.

Methods for solving systems of linear equations. Gaussian elimination and variants. Cholesky-Banachiewicz decomposition. LU decomposition. $PA = LU$ decomposition. $PAQ = LU$ decomposition. Conditioning of linear equation systems. Iterative methods for solving linear systems. Convergence of iterative methods. Termination conditions. Jacobi method. Gauss-Seidel method.

Matrix eigenvalue localisation. The Gershgorin circle theorem theorem.

Numerical differentiation. Obtaining better approximations. Asymmetric formulas. Richardson extrapolation.

Least squares approximation.

Physics 1

Mechanics. Kinematics. Newton's laws of motion. Conservation of momentum. Galilean transformation. Work and mechanical energy. Kinetic and potential energy. Conservative forces. Conservation of angular momentum. Motion in non-inertial reference frames. Motion of a rigid body. Moment of inertia. Law of gravitation. Small oscillations. Harmonic oscillations. Damped oscillations. Mechanical resonance. Lorentz transformation. Relativistic dynamics. Principle of least action. Lagrange's equations. Integrals of motion. Motion in a central field. Kepler's laws. Hamilton's equations.

Electromagnetism. Coulomb's law. Electrostatic field. Electrostatic potential. Electric

dipole. Gauss's theorem. Dielectric polarisation. Conductor in a field. Electric capacitance. Parallel-plate capacitor. Current continuity equation. Electromotive force. Differential form of Ohm's law. Field energy. Magnetic field of a moving charge. Biot-Savart law. Lorentz force. Ampère's law. Circuit with current in a magnetic field. Magnetic properties of media. Vector properties of the magnetic field. Electromagnetic induction. Self-induction. Maxwell's equations. Electrical oscillations. Wave equation.

Probability

Probability space. General definition of probability. Random events and their description. Classical and geometric probability. Conditional probability and independence of events. Random variables, random vectors, and their distributions. Cumulative distribution functions. Discrete, absolutely continuous, and mixed distributions. Overview of the most important probability distributions. Functions of random variables and vectors. Expected value. Variance and covariance matrix. Moments and quantiles. Independence of random variables. Various types of convergence of random variables. Laws of large numbers and their applications. Weak convergence of distributions. Central Limit Theorem for independent random variables and its applications.

Language Course 3 & 4

Physical Education and Sports 2

Semester 4

Research Workshop 1

Work organisation and team collaboration. Preparation and monitoring of the work schedule. Problem definition. Literature review concerning the analysed field. Preparation of research reports. Preparation of presentations for the obtained results.

Databases

Database Management Systems (DBMS). Relational databases. Normalisation and the problem of data redundancy. Ensuring data integrity—referential integrity, primary key uniqueness. Database design. Entity-Relationship Diagrams (ERD).

SQL language. Data selection and modification. Transaction processing, transaction isolation, distributed transactions. Lock issues and isolation management. Database server programming: stored procedures, views, triggers. Performance optimisation: indexes, statistics and execution plans, performance monitoring methods.

Selected topics in Data Warehousing and Business Intelligence systems. Creating database applications—design principles and basic architectural solutions. Database access from Java and .NET applications: establishing connections, efficient execution of SQL commands. Security issues and selected aspects of database server configuration using Oracle RDBMS as an example. Oracle RDBMS architecture: databases, instances, physical data organisation management (tablespaces, files), network access configuration, data protection, and recovery mechanisms. Introduction to Big Data storage and non-relational solutions using Apache Hadoop as an example. Introduction to NoSQL platforms. Spatial data management.

Introduction to Machine Learning

Preliminary data processing: dimension reduction, feature extraction: principal component analysis (PCA), SVD, multidimensional scaling, feature correlation. Unsupervised learning: cluster analysis. Distance measures, criterion function optimisation algorithms (k-means, k-medoids), hierarchical methods. Generalisation. Overfitting. Cross-validation. Object similarity measures, data segmentation based on similarity. Comprehensive analysis of all stages of construction and application of data analysis tools on selected examples (case studies) using the CRISP-DM methodology.

Binary/multi-class classification problem: Bayesian approach, empirical Bayesian classifiers (Naive Bayes, k-NN classifier). Linear classification methods: LDA, logistic regression, SVM (linear version). Trees as classifiers. Rule extraction from trees. Association rules, rule-based systems. Evaluating classifier effectiveness. Confusion matrix. Error measures and their interpretation.

Mathematical Statistics

Introduction to statistics. Descriptive statistics and basics of data exploration. Subject matter and methodology of statistics. Basic terminology. Graphical methods for data presentation and numerical characteristics of samples.

Foundations of statistical inference. Statistical model. Empirical distribution function. Fundamental theorem of mathematical statistics (Glivenko-Cantelli lemma). Sufficient statistics and factorisation criterion. Exponential families of distributions.

Point estimation. Basics of point estimation theory; mean squared error. Unbiased estimators and minimum variance unbiased estimators (MVUE). Cramér-Rao bound; estimator efficiency. Consistency of estimators. Methods of constructing estimators: method of moments, quantiles, maximum likelihood, EM method, least squares, L-estimators, M-estimators, Bayesian estimators. Standard error of an estimator; jackknife and bootstrap methods.

Interval estimation. Definition, interpretation, and properties of confidence intervals. Methods of constructing confidence intervals. Basic confidence intervals. Determining sample size for interval estimation with a given level of significance.

Hypothesis testing. Basic concepts of hypothesis testing: types of hypotheses, Type I and Type II errors, test power, significance level, and test size. Uniformly most powerful tests: Neyman-Pearson lemma; testing composite hypotheses in families with monotone likelihood ratio (Karlin-Rubin theorem); unbiased tests. Examples of test construction. Basic parametric tests for one sample and two samples. Likelihood ratio tests. Goodness-of-fit testing: chi-square test, Kolmogorov-Smirnov test, and other tests based on empirical distribution; normality tests. Contingency tables and chi-square independence test.

Physics 2

Thermodynamics. First law of thermodynamics. Ideal gas processes. Carnot cycle. Second law of thermodynamics. Entropy. Kinetic theory of gases. Barometric formula. Equipartition theorem. Temperature and statistical entropy. Maxwell velocity distribution. Optics. Fermat's principle. Refractive index. Prism. Lens. Interference. Fresnel diffraction. Light polarisation. Doppler effect. Microphysics. Photoelectric effect. Compton effect. X-rays. de Broglie waves. Rutherford scattering experiments. Bohr atomic model. Franck-Hertz experiment. Schrödinger equation. Heisenberg's indeterminacy principle. One-dimensional potential well. Pauli exclusion principle. Quantum numbers of hydrogen. Properties of nuclei in the ground state. Liquid drop model. Radioactivity. Nuclear fission and fusion reactions.

Elective 1**Language Course 5 & 6****Physical Education and Sports 3**

Semester 5

IT Systems Engineering

Preliminary concepts: programming as an engineering process, introduction to the concepts of teamwork and project scalability. Basic concepts of object-oriented programming: classes, methods, inheritance, polymorphism, overloading, generalisation, specialisation, examples of simple UML diagrams. Advanced concepts of object-oriented programming: meta-classes, threads, nuances of UML diagrams. Software development models: waterfall, spiral, pyramid, XP (Extreme Programming), Open-Source. Requirements for software developers. Requirements for the final product.

Methods of Computational Statistics

Basic matrix decompositions and their application in data analysis (PCA, pseudo-inverse), matrix norms. Simple and multiple linear regression models: Ordinary Least Squares (OLS) method, properties of the OLS estimator, basic tests, goodness-of-fit diagnostics. Lasso and Ridge regression methods, Principal Component Regression (PCR).

Parametric non-linear regression models and the non-linear least squares method. Non-parametric estimation of the regression function: moving average and locally linear estimators. Binary regression models: logistic and probit regression. Maximum likelihood estimators in the logistic model and their properties. Multiclass logistic regression. Applications in classification.

Basic characteristics of stationary time series: autocovariance function, partial autocorrelation function, cumulative distribution, and spectral density. Time series models: autoregressive (AR), moving average (MA) and ARMA processes. The problem of optimal linear forecasting and its solution. Decomposition of non-stationary time processes.

Optimisation Methods

Examples and classification of optimisation problems. Global and local solutions of unconstrained optimisation problems. Optimality conditions (necessary and sufficient) for unconstrained optimisation problems. Convex sets and functions and their significance in optimisation. Descent methods (with line search) and trust region methods for unconstrained optimisation problems. General convergence conditions for descent methods. Convergence rates of optimisation methods.

Conjugate gradient method. Newton's method and quasi-Newton methods. Linear least squares problem. QR decomposition and normal equations methods for solving the linear least squares problem. Gauss-Newton method for least squares problems. Newton's method for solving systems of nonlinear algebraic equations. Optimality conditions for constrained optimisation problems: Karush-Kuhn-Tucker conditions; sufficient optimality conditions; constraint qualification conditions. Dual problems for convex constrained optimisation problems.

Linear programming problem. Simplex method. Interior point methods for solving linear programming problems. Introduction to solving constrained optimisation problems: penalty function method; augmented Lagrangian method; exact penalty function method. SQP (Sequential Quadratic Programming) methods; variable elimination methods; methods for solving quadratic programming problems.

Examples of integer optimisation problems. Optimality conditions for integer optimisation problems; linear programming relaxation; Lagrangian relaxation. Complexity of integer optimisation problems. Branch and bound method for integer optimisation problems. Cutting-plane method and column generation method for integer optimisation problems.

Stochastic Processes

Markov Chains. Time-homogeneity. Chapman-Kolmogorov equations. Recurrent and transient states, communicating class. Stationary and limiting distributions; conditions for existence and uniqueness. Ergodicity, detailed balance, the Metropolis–Hastings algorithm.

Poisson Processes. Poisson and Lévy processes. Interarrival times distribution (exponential) and properties of jumps. Construction methods for Poisson processes. Thinning, superposition, and non-homogeneous Poisson processes. Compound Poisson processes and extensions.

Markov Chains with Continuous Time. Definition and properties of continuous-time Markov chains. Generator (intensity) matrix and dynamics description. Chapman-Kolmogorov equations for continuous time. Embedded chains and exponential waiting times. Limiting distributions and convergence results.

Gaussian Processes. Canonical representation. Multivariate Gaussian vectors and covariance matrices. Gaussian and Wiener processes. Finite-dimensional distributions and covariance function properties. Applications of Gaussian processes in machine learning.

Web Application Programming

Documents in HTML standard, forms. Document formatting: CSS and JavaScript. XML standard: DTD, XML Schema, XSLT. Specifications of HTML/XHTML standards. Communication with servers: HTTP, AJAX, JSON. Server-side programming: basics of HTTP (GET/POST, ENCTYPE, data security, declaring Content-Type in scripts). Java Servlets. Python web scraping.

Computer Networks (elective block)

Elective 2

Semester 6

Research Workshop 2

Work organisation and team collaboration. Preparation and monitoring of the work schedule. Problem definition. Literature review concerning the analysed field. Preparation of research reports. Preparation of presentations for the obtained results.

Computational Intelligence Methods in Data Analysis

Neural networks. Universal approximation theorem. Neural network training. Stochastic Gradient Descent. Gradient-based optimisation methods. Hopfield networks. Kohonen networks. Overfitting detection and prevention. Evolutionary algorithms. Fuzzy logic and fuzzy inference. Hybrid approaches: neuro-evolution and neuro-fuzzy inference. Swarm algorithms. Ant Colony Optimisation, Ant Colony System, Ant Miner, Particle Swarm Optimisation, Artificial Bee Colony. Solving games and dynamic control processes. Reinforcement learning.

Data Warehouses and Business Intelligence

Introduction to data warehouses and Business Intelligence systems. Designing a data model for a data warehouse; data model versus normalisation. Dimensional modelling techniques. The role of the time dimension and data changes in data model design. Data integration using ETL/ELT tools. The use of OLAP data cubes in reporting. Overview of reporting and data visualisation methods provided by selected Business Intelligence platforms. Implementation of sample fact and dimension tables. Application of advanced modelling techniques, e.g., slowly changing dimensions. Preparation of a data cube and visualisation of data from fact tables and related dimensions. Analysis of the role of metadata in configuring a Business Intelligence system. Configuration of reports and data visualisations in a Business Intelligence system

Interdisciplinary Project

Methods of team building. Methods for evaluating IT project concepts with regard to their relation to competing products on the market. Assumptions of methodologies used in preparing and/or implementing innovative projects, e.g., Problem-Based Learning, Design Thinking. Collaboration with central university units and/or external investors to prepare a commercialisation path for the results of individual projects.

IT Systems Architecture

Client-server and multi-tier architecture. Modularisation, communication, and distributed systems. Fundamentals of web application architecture. Web services: SOA/SOAP and ROA/REST. Virtualisation and Cloud Computing. Containerisation (Docker). Orchestration and cloud-native architectures (Kubernetes). Selected architectural patterns and information system architecture design. High availability and high performance. IoT architecture and protocols. Fundamentals of OS theory and Linux architecture. MS Windows architecture.

Elective 3

Elective 4

Semester 7

Data Storage in Big Data Systems

Overview of Big Data issues and software, including projects such as Apache Hadoop. Big Data systems architecture: data acquisition, storage, and data transfer in a distributed system architecture; data analysis.

Data acquisition from diverse sources using Big Data platforms. Batch processing vs stream processing. Programming solutions using Big Data platforms, taking into account the use of parallel and distributed data processing and environments that facilitate the creation of data processing code in batch and stream modes.

Design and architectural patterns, such as Lambda and Kappa architectures. Implementing data filtering and aggregation methods. Selection of data formats and structures for stored data.

Data Transmission

Internet Topology. OSI model, Ethernet, 10BaseT, WLAN.

Link Layer: Ethernet, VLANs, ATM and Carrier Ethernet.

IP Routing: Longest prefix match, IP addressing, Dijkstra's Algorithm.

Interior Gateway Protocols (IGP): RIP, OSPF. IP/MPLS, BGP.

L3 VPN. TCP. Content switching, routing.

Diploma Seminar

The objective of the course is to prepare students for their diploma thesis defence by monitoring current progress in its preparation and conducting practical exercises related to the presentation of the thesis topic, its implementation process, and the obtained results.

Group Project

Consolidation of principles for industrial software/application development. Improvement of software development skills through creating applications in a small team (2–3 persons). Conducting unit tests and integration tests. Preparation of a complete documentation set: work schedule; requirements specification (business documentation); architectural documentation; technical documentation; acceptance test plan; user manual; change log.

Introduction to Financial Management

Financial mathematics. Different types of interest rates. Time value of money. Bonds. Options. CRR binomial model. LIBOR. Macroeconomic accounts.

Diploma Thesis