

V. N. Karazin Kharkiv National University
Department of Theoretical and Applied Computer Science

Student's Handbook

Bachelor of Computer Science

Preamble

In this handbook the students find the student's educational plan and annotations to all its units. The units are grouped: core units are obligatory, they include fundamental mathematics and computer science units; selective units suppose the free choice of student to take any of proposed ones, the number of selective units to pass in each semester is given in the table below. There are some mathematically oriented selective units but the majority of them are computer science oriented and cover the areas of modern IT technology. At the end of studies student defend their diploma works.

As a result of this educational programme the students gain the following competences.

Knowledge and understanding:

- the ability to develop and analyze mathematical models, to justify methods and approaches to solving theoretical and applied problems in computer science;
- the ability to make logical conclusions, to use formal languages and models of algorithmic computing, to design, develop and analyze algorithms with evaluation of their efficiency and complexity, to determine the solvability and unsolvability of algorithmic problems;
- the ability to apply theoretical and practical fundamentals of modeling methods and techniques for analyzing characteristics and behavior of complex objects and systems; to conduct computing experiments and analyze their results;
- the ability to develop multilevel computing models including databases, knowledge bases, data storages and distributed processing of big data;
- the ability to perform intellectual data analysis, operational processing of data and visualizing its results in the process of solving applied problems.

Application of knowledge and understanding:

- effectively use modern mathematical apparatus in professional activities to solve theoretical and applied problems in the process of analysis, synthesis and design of information systems in industries;
- use formal models of algorithms and computable functions, determine the solvability, partial solvability and unsolvability of algorithmic problems; design, develop and analyze algorithms, evaluate their efficiency and complexity;
- design and develop software using different programming paradigms such as generalized, object-oriented, functional, and logical ones with corresponding models, methods and computing algorithms, data structures and control mechanisms;

- implement computing on the base of cloud services and technology, parallel and distributed computing.

Forming judgments:

- formulate, analyze and solve problems related to computer science in the process of modeling and software development.

EDUCATIONAL PROGRAMME

Mathematics Core	10
Mathematical Analysis.....	10
Elements of Algebra and Number Theory	11
Elements of Mathematical Logic and Discrete Mathematics.....	11
Analytic geometry.....	12
Linear Algebra	12
Discrete Mathematics.....	13
Differential Equations	14
Probability Theory and its Applications.....	14
Introduction to Mathematical Statistics.....	15
Methods of Optimization and Operations Research.....	16
Computer Science Core	16
Programming 1	17
Programming 2	17
Algorithms and Data Structures.....	17
Architecture of Computing Systems	18
Object-oriented Programming (C++ Language).....	19
Object-oriented Programming (Java Language)	19
Operating Systems.....	20
Mathematical Logic and Logic Programming	21
Information Networks.....	21
Introduction to SQL Databases	22
Theory and Methods of Relational Databases Design.....	22
Web-programming Technology	22
Declarative Programming (Functional Languages).....	23
Methods of user interface design.....	24
Introduction to Parallel Processes Programming (C ++ and Java Languages).....	24
Development of Compilers for Domain Specific Languages.....	25
Parallel and Distributed Computing	25
Introduction to Artificial Intelligence.....	25

Mathematics Electives	26
Computing Theory	26
Automata Theory	26
Introduction to Numerical Methods.....	27
Mathematical Basics of Computer Graphics	27
Analytical Methods of Geometric Modelling.....	28
Mathematical Methods of Image Processing.....	28
Computer Science Electives	29
Introduction to programming with Python	29
Introduction to Programming with .NET Platform	29
Introduction to Programming with JavaScript.....	30
Version Control with Git	30
Programming with Python	31
Algorithms of Computational Geometry	31
Programming with C#	32
Information Theory and Coding.....	32
Methods and Technologies of Big Data Processing.....	33
Object-oriented Programming with C#	34
Advanced Scala Programming	34
Color and Light Models in Computer Graphics.....	35
Modern Java technologies	35
Introduction to Web-development with Java	36
IoT Systems Construction for Enterprises and Organizations	36
Categories Theory in Computer Science.....	37
Web-programming with Java.....	37
Methods of Computer Networks Management	37
Introduction to Machine Learning.....	38
Multithread Programming and Web-development with .NET platform.....	39

Educational Programme

Code	Unit Name	ECTS Credits	Credit\exam	Semester
CU01	History of Ukraine	3	Exam	1
CU02	Philosophy	3	Exam	4
CU03	English Language	10	Exam /credit	1,2,3
CU04	Professional English	3	Exam	5
CU05	Mathematical Analysis	18	Exam	1,2,3,4
CU06	Elements of Algebra and Number Theory	6	Exam	1
CU07	Elements of Mathematical Logic and Discrete Mathematics	4	credit	1
CU08	Analytic Geometry	4	credit	1
CU09	Linear Algebra	4	credit	2
CU10	Discrete Mathematics	12	Exam	2,3,4
CU11	Differential Equations	4	Exam	5
CU12	Probability Theory and its Applications	8	Exam	5,6
CU13	Introduction to Mathematical Statistics	4	Exam	7
CU14	Programming	10	Exam	1,2
CU15	Object-oriented Programming (C++ Language)	4	Exam	3
CU16	Object-oriented Programming (Java Language)	4	Exam	4
CU17	Algorithms and Data Structures	4	Exam	3
CU18	Architecture of Computing Systems	4	credit	2

CU19	Operating Systems	4	credit	3
CU20	Mathematical Logic and Logic Programming	4	credit	4
CU21	Information Networks	4	credit	4
CU22	Methods of Optimization and Operations Research	8	Exam	6,7
CU23	Introduction to SQL Databases	4	Exam	5
CU24	Web-programming Technology	4	credit	5
CU25	Declarative Programming (Functional Languages)	4	credit	5
CU26	Theory and Methods of Relational Databases Design	4	Exam	6
CU27	Methods of User Interface Design	3	Exam	6
CU28	Introduction to Parallel Processes Programming (C ++ and Java Languages)	4	Exam	6
CU29	Development of Compilers for Domain-oriented Languages	3	credit	7
CU30	Software Systems Design	4	Exam	7
CU31	Parallel and Distributed Computing	4	Exam	7
CU32	Course Scientific Research	3	credit	7
CU33	Introduction to Artificial Intelligence	4	Exam	8
CU34	Technical English	3	credit	8
CU35	Research Practice	5	credit	8
CU36	Preparation of Diploma	2	defense	8
Core Units Summary (ECTS)		180		
Selective Units				
SU 2.1.1	InterFaculty Unit 1	3	credit	
SU 2.1.2	InterFaculty Unit 1	3	credit	
SU 2.1.3	InterFaculty Unit 1	3	credit	
SU 2.1.4	InterFaculty Unit 1	3	credit	

SU 2.2.1	Unit 1 (Elective disciplines)	4	credit	2
	Introduction to Programming with Python			
	Introduction to Programming with .NET Platform			
	Introduction to Programming with JavaScript			
	Version Control with Git			
SU 2.2.2	Unit 2 (Elective disciplines)	4	credit	3
	Computing Theory			
	Programming with Python			
SU 2.2.3	Unit 3 (Elective disciplines)	4	credit	4
	Automata Theory			
	Algorithms of Computational Geometry			
	Programming with C#			
SU 2.2.4	Unit 4 (Elective disciplines)	4	credit	5
	Mathematical Basics of Computer Graphics			
	Information Theory and Coding			
	Methods and Technologies of Big Data Processing			
SU 2.2.5	Unit 5 (Elective disciplines)	4	credit	6
SU 2.2.6	Unit 6 (Elective disciplines)	4	credit	6
	Object-oriented Programming with C#			
	Advanced Scala Programming			
	Color and Light Models in Computer Graphics			
	Modern Java technologies			
SU 2.2.7	Unit 7 (Elective disciplines)	4	credit	7
SU 2.2.8	Unit 8 (Elective disciplines)	4	credit	7
	Introduction to Web-development with Java			

	Mathematical Methods of Image Processing			
	IoT Systems Construction for Enterprises and Organizations			
SU 2.2.9	Unit 9 (Elective disciplines)	4	credit	8
SU 2.2.10	Unit 10 (Elective disciplines)	4	credit	8
SU 2.2.11	Unit 11 (Elective disciplines)	4	credit	8
SU 2.2.12	Unit 12 (Elective disciplines)	4	credit	8
	Categories Theory in Computer Science			
	Introduction to Numerical Methods			
	Analytical Methods of Geometric Modelling			
	Web-programming with Java			
	Methods of Computer Networks Management			
	Introduction to Machine Learning			
	Multithread Programming and Web-development with .NET platform			
	Selective Units Summary (ECTS)	60		
	Units Summary (ECTS)	240		

Mathematics Core

Name	Mathematical Analysis
Programme	bachelor
Semester	1-4
ECTS credits , Exam / Credit	18 ECTS, Exam 1 semester 2 hours lectures and 4 hours practical work per week 2 semester 2 hours lectures and 2 hours practical work per week 3 semester 2 hours lectures and 2 hours practical work per week 4 semester 2 hours lectures and 2 hours practical work per week
Short description	The objective of the course is to present an introduction to classical mathematical analysis, to elementary complex analysis and to elementary functional analysis.
Topics	<i>Semester #1: 2 hours of lectures and 4 hours of practical work per week</i> 1. The real field. Convergent sequences. The number e. Series. The root and ratio tests. Absolute convergence. Power series. 2. Limits of functions. Continuous functions. Discontinuities. 3. Differentiation. Derivatives of higher order. <i>Semester #2: 2 hours of lectures and 2 hours of practical work per week</i> 4. Taylor's theorem. Taylor's series. Convex functions. 5. Infinite products. 6. The Riemann integral. Integration and differentiation. 7. Improper integrals, Beta and Gamma functions. <i>Semester #3: 2 hours of lectures and 2 hours of practical work per week</i> 8. Functions of several variables. 9. Double and triple integral. Change of variables. 10. Vector Analysis. <i>Semester #4: 2 hours of lectures and 2 hours of practical work per week</i> 11. Functions of a complex variable, domains and curves in the complex plane. Cauchy-Riemann conditions. Holomorphic functions. The integral of functions of a complex variable function and the Cauchy theorem. The Cauchy integral formula and its application. 12. Hilbert spaces, Cauchy-Schwarz inequality, orthoprojectors, orthonormal bases. General form of linear functional. 13. Fourier series. 14. Banach spaces. Linear operators. Norm of operator. Convergence in norm and pointwise convergence. Linear functionals.

Name	Elements of Algebra and Number Theory
Programme	bachelor
Semester	1
ECTS credits, Exam / Credit	6 ECTS credits, Exam, 3 hours lectures and 3 hours practical work per week
Short description	The objective of the course is to present an introduction to elementary number theory and to algebra of complex numbers and polynomials of one variable.
Topics	Basic number theory: divisibility, greatest common divisor, linear Diophantine equations, prime numbers, unique factorization into primes, residue classes, Chinese remainder theorem, Fermat's little theorem, Euler's theorem. Complex numbers: definition, geometric interpretation, trigonometric form, De Moivre's formula, roots of unity, solving cubic and quartic equations. Polynomials of one variable: definition, degree, greatest common divisor, irreducible polynomials, unique factorization into irreducibles, roots of polynomials, number of roots, interpolation, root multiplicities, derivative (with application to multiple roots), partial fractions.

Name	Elements of Mathematical Logic and Discrete Mathematics
Programme	bachelor
Semester	1
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The aim of the course is to prepare students to learn mathematics at the university level. We suggest an informal preparatory discussion of the basic concepts and examples from set theory, mathematical logic, and discrete mathematics. The course focuses on solving and discussing simple introductory problems.
Topics	Elements of set theory. Operations with sets. Elements of mathematical logic. Propositions and connectives. Predicates and quantifiers. Operating with quantifiers. Mathematical proofs: examples of theorems and their proofs. Mathematical induction. Relations: equivalence and partial order. Elements of combinatorics. Permutations and combinations, binomial theorem, Pascal's triangle. Elements of graph theory.

Name	Analytic geometry
Programme	bachelor
Semester	1
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students to work with vectors in two-dimensional and three-dimensional affine spaces, to find scalar, vector and mixed product of vectors, to find the equations of lines and planes in different forms, solve various problems related to these objects.
Topics	1. Affine space. Affine coordinate system. Transformation of the affine coordinate system Orientation. Actions with vectors. 2. Scalar product 4. Vector product. Mixed product 5. Different types of equations of a straight line on a plane 6. The relative location of two straight lines on a plane, the distance from a point to a straight line 7. Different types of equations of a straight line and a plane in space 8. The relative location of lines and planes 9. The distance from a point to a straight line, from a point to a plane, the distance between straight lines

Name	Linear Algebra
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	The course consists of two modules: linear algebra and analytic geometry. The module of linear algebra contains the notion of a matrix, matrix operations; the definition and methods of computation of the determinant of a square matrix; the notion of a real linear space, its basis and dimension; the general theory of linear systems; quadratic and bilinear functionals and forms; linear operators. The module of analytic geometry contains: the canonical equations and geometric characteristics of the curves and the surfaces of the second order, reduction of a general equation of the curve and the surface of the second order to the canonical form.
Topics	1. Notions of fields, rings, groups. Residue rings and fields. 2. Determinants. Properties of determinants. Methods of calculating determinants. 3. Matrices, operations with matrices. Inverse matrix. Matrix rank. 4. Systems of linear equations. Kramer's rule. Gauss method of solving a system of linear equations. General theory of systems of linear equations

	5. Bilinear functionals and forms. Quadratic functionals and forms. Lagrange's method of reducing the quadratic form to the canonical form. 6. Linear operator. Linear operator matrix. Eigennumbers and eigenvectors of a linear operator. Invariant subspaces. Diagonalization of the linear operator matrix. 7. Orthogonal diagonalization of quadratic forms. 8. Canonical theory of curves of the second order. Definitions and main characteristics of second-order curves (ellipse, hyperbola, parabola). The equation of the tangent line to the curve of the second order. 9. Canonical theory of surfaces of the second order. Some methods of forming surfaces, their equations. Definition and properties of second-order surfaces (ellipsoids, hyperboloids, paraboloids, cones and cylinders). Rectilinear generators on second-order surfaces. The tangent plane of the surface of the second order. 10. Reduction a general equation of a curve and surface of the second order to the canonical form. Finding the canonical coordinate system.
--	---

Name	Discrete Mathematics
Programme	bachelor
Semester	2,3,4
ECTS credits , Exam / Credit	12 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	Discrete Mathematics is a fundamental course that equips students with the necessary knowledge and skills to work effectively with data structures and algorithms. The course delves into various essential topics, including but not limited to combinatorics, graphs, codes and cryptography. Throughout the course, students will engage in hands-on practical work using Python programming language.
Topics	Semester 2 Basic combinatorics Enumerative combinatorics Graph Theory Semester 3 Binary relations Group theory Basic Number Theory Basic Cryptography Basic Information theory Semester 4 Applications of linear algebra methods to combinatorics and graph theory Error-correcting codes

Name	Differential Equations
Programme	bachelor
Semester	5
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	The course is devoted to the classical theory of ordinary differential equations and its applications. Methods to solve various types of differential equations and linear systems of differential equations are considered. The students also learn theorems of existence and uniqueness of the solution of Cauchy problems and the basics of Lyapunov's stability theory.
Topics	Applied problems that lead to ordinary differential equations. Introduction to the theory of ordinary differential equations. Integrable classes of differential equations of the first order. Theory of linear differential systems and equations with continuous coefficients. Theory and methods to solve linear differential equations and systems with constant coefficients. Theorems of existence and uniqueness of the solution of Cauchy problems. Basics of Lyapunov's stability theory.

Name	Probability Theory and its Applications
Program	bachelor
Semester	5, 6
ECTS credits , Exam / Credit	8 ECTS, Exam, 2 hours lectures, 2 hours practical work per week
Short description	Semester 5 The course is designed to teach students the methods of statistical theory including its basics, the tasks and the methods, namely: the classical concept of probability and its direct calculation by default; finding the probability of an event using operations on the events and properties of probability; the concept of a random variable, the distribution and the numerical characteristics of a random variable, the calculation of the probabilities of various events by using random variables; distribution laws and the numerical characteristics of some random variables often encountered in practice. Semester 6 Acquaintance of students with theoretical and practical knowledge of probabilistic modeling of real phenomena and processes, in particular, with the main types of random processes and event flows, elements of stochastic analysis of random processes (continuity, differentiation, integration), basics of mass service theory.

Topics	Semester 5 The subject, problems and the basic concepts of probability theory Classical, geometric and axiomatic definition of the probability of a random event Actions on random events and properties of their probabilities Conditional probability Test repetition Random variable, probability distribution law for random variable values, distribution function and distribution density Basic properties of the distribution function and the probability density, numerical characteristics of a random variable Basic discrete distributions: binomial, geometric, Poisson Basic continuous distributions: uniform, exponential, normal, Rayleigh Random vector and its numerical characteristics Functions of random variables Semester 6 Elements of stochastic modeling, Monte-Karlo method Basic types, classification of random processes, Markov Chains Markov processes with discrete states and continuous time Mass service systems Basics of stochastic analysis Markov processes with continuous states Basics of statistics of random processes Branching processes Event streams and their classification

Name	Introduction to Mathematical Statistics
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	The course teaches students the basics, tasks and methods of mathematical statistics, namely: tasks of estimating unknown parameters of the general population, constructing sampling distribution functions, testing statistical hypotheses using Pearson's, Kolmogorov's agreement criteria, Romanovsky's rule, familiarization with correlation and regression analysis, familiarization with the concept of random processes and sequences, with correlation theory, with stationary random processes and sequences and their application in computer science problems.

Topics	1. The subject, problems and basic concepts of mathematical statistics. 2. Statistical models. 3. Methods of estimating unknown parameters. 4. Testing of statistical hypotheses. 5. Elements of correlation and regression analysis. 6. Elements of dispersion analysis. 7. Definition and main properties of random functions. 8. Correlation theory of random functions. 9. Elements of analysis of random functions. 10. Stationary random processes. 11. Stationary random processes and sequences.
---------------	--

Name	Methods of Optimization and Operations Research
Programme	bachelor
Semester	6, 7
ECTS credits , Exam / Credit	8 ECTS, Exams, 2 hours lectures and 2 hours practical work per week
Short description	In this course, students learn methods to solve various optimization problems in finite-dimensional spaces and their applications. Extrema of multi-variable functions and linear programming problems are considered. With the application of the theory, transportation problems and some problems of game theory are solved.
Topics	Semester 6. Optimization problems in finite-dimensional spaces. Necessary and sufficient conditions of extrema of multi-variable functions. Method of Lagrange multipliers. Applied problems that lead to linear programming problems. Simplex method to solve the linear programming problem. Dual problems of linear programming. Methods to solve different transportation problems. Hungarian algorithm and its applications. Transportation networks. Ford-Fulkerson algorithm. Semester 7. Game theory. Methods to solve antagonistic games of different orders and their relations with linear programming problems. Symmetric games and their relation to linear programming. Basics of non-antagonistic games. Nash equilibrium and Pareto optimality. Methods to solve bimatrix games of different orders. Basics of positional games.

Computer Science Core

Name	Programming 1
Programme	bachelor
Semester	1
ECTS credits , Exam / Credit	4 ECTS, exam, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students basics of algorithms and fundamental methods and patterns of structured programming such as “functional decomposition” and “divide and conquer”. Basic data types and their effective use are discussed and applied in problems solving. Students use C language for practical work.
Topics	Basic structures of algorithms, flowcharts construction, specification of problems and structuring information. Simple data types and their computer representation, operations and operators in C programming. Functions, their purposes and use, making the solutions modular and reusable. The concept of pointer, address operations and pointer arithmetic. Parameters of functions: by value and by reference. The concept of array, computer representation of arrays, static and dynamic arrays, multidimensional arrays.

Name	Programming 2
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	6 ECTS, exam, 2 hours lectures and 4 hours practical work per week
Short description	The course is teaching students working with composite data types and their effective use in solving problems with structured information with variable part. File as data type and working with text and binary files on different levels are considered. Students use C language for practical work.
Topics	Enumerations, structures and unions, their computer representation, access to elements, application to problem solving. Combinations of composite data types (nested structures, arrays of structures, arrays and strings as elements of structures and unions, etc.) for effective structuring of information in problems solving. Concept of file, text files, binary files, concept of buffering. Operations with files: upper level vs. low level access. Random access to files. Persistence. Course project.

Name	Algorithms and Data Structures
Programme	bachelor
Semester	3

ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students to select the effective data structures working with algorithms. Such data structures as linked lists, double lists, trees and graphs are considered. Classic algorithms of search and sorting as well as basic algorithms on graphs are discussed and implemented. Students use C or C++ language for practical work.
Topics	Sequential and random access to data, comparison arrays with linked data structures, evaluating the performance of basic operations. Working with all sorts of linked lists and trees including AVL trees. Concept of ADT (Abstract Data Type) and classification of ATD. Effective implementation of ATD with data structures. Graphs, their possible computer representations with different data structures, basic algorithms on graphs. Search algorithms (sequential, binary, using trees, backtracking, etc.) Classic sorting algorithms with n^2 and $n \log n$ performance including simple ones (bubblesort, shakersort, insertion, etc.) and advanced ones (Shellsort, heapsort, quicksort, mergesort, etc.).

Name	Architecture of Computing Systems
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, Credit, 1 hour lectures and 2 hours practical work per week
Short description	The task of the discipline is the formation in students' principles of computer organization, computer interfaces, the basics of assembly language programming, working with binary data and numbers in binary form, working with dynamic memory, the basics of structural programming on Assembler language.
Topics	During the lectures: Basic concepts about computers. Data formats in computers. Basic functional elements of the computer. Program model of the processor. Multi-core processor. Virtual cores. Register sets. Status and control registers. Addressing methods and address arithmetic. Building multi-module applications on Assembler language. The architecture of the graphics core, in memory. Register spin processor x87. MMX extension. XMM block. During the practical work: Translation of numbers from one numbering system to another, coding of

	information in computers, execution of bit logical operations and construction of additional binary codes of negative numbers. Bitwise operations in assembly language. Operations on bits and bytes. Shift operations in assembler language. Calculation of arithmetic expressions Organization conditional jump. Strings processing. Organization of cycles and work with arrays. Procedures. Call, transmission of parameters.
--	--

Name	Object-oriented Programming (C++ Language)
Programme	bachelor
Semester	3
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	This course learns how to write C++ programs using Object-Oriented Programming concepts such as objects, classes, inheritance and polymorphism. Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism, etc. in programming. The main aim of OOP is to bind together the data and the functions that operate on them so that no other part of the code can access this data except that function.
Topics	Class definition, its properties and behavior. Constructors and destructors. Encapsulation, inheritance and polymorphism. Basic OOP concepts in the C++ language. Multiple inheritance. Overloading operations in the C++ language. Templates of classes and functions. Input/output streams in the C++ language. STL library.

Name	Object-oriented Programming (Java Language)
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	This course learns how to create extensible Java applications using interfaces, classes and composition, advanced java programming using exception handling, multithreading. This course ensures a common foundation for Object-oriented Programming in Java. After course students can develop java applications using interfaces

	and OOP concepts.
Topics	Basic concepts of OOP in Java. Inheritance and composition. Concepts of interface, internal class, types of internal classes and tasks requiring their use. Collections. Input and Output streams. Multithreading. Reflection API.

Name	Operating Systems
Programme	bachelor
Semester	3
ECTS credits , Exam / Credit	4 ECTS, Credit, 1 hours lectures and 2 hours practical work per week
Short description	The task of the discipline is the formation in students of basic system concepts and skills, a holistic vision of the modern level of the main characteristics of computer system software (software), which are clearly reflected in programs and must be taken into account when developing and executing programs: principles, methods and tools for software development and means of its improvement; methods of controlling external devices and methods of memory manipulation; strengthening of interdisciplinary ties, development of systems thinking, without which the effective use of information technologies is impossible.
Topics	During the lectures, students will learn: Architecture of operating systems. Management of processes and data flows. Management of processes and data flows – task planning. Management of processes and data flows – synchronization of tasks. Management of RAM. RAM management in x86 architecture processors. RAM management in Linux and Windows OS. Input-output control. Input-output control in Linux, UNIX, Windows OS. File systems. Implementation of file systems. Concept of distributed information processing. Remote procedure call (RPC). During the practical work: MS DOS operating system features. Virtualization. Virtualization support systems VMWare, Oracle VirtualBox Operating system Microsoft Windows. Installation. Windows OS administration. PowerShell scripting language – output, processes, variables.

	PowerShell scripting language – working with files, objects (WMI, COM), logs. Organization of Batch Files and Scripts in Windows OS. Linux. Installation. UNIX structure. commands. Linux. File system. Access rights. Processes. The simplest scripts using Bash. Stream redirection using Bash.
--	--

Name	Mathematical Logic and Logic Programming
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course covers propositional calculus, predicate calculus, and resolution calculus. Students learn to manipulate logical statements, construct formal proofs, and apply automated reasoning. Practical focus on Prolog language enables solving problems using logic-based programming.
Topics	Propositional logic and propositional calculus Predicates and first-order logic Resolution calculus Formal proofs and automated theorem proving Logic-based programming with Prolog

Name	Information Networks
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	A modern complex technical system is, as a rule, a system with intensive use of computer networks. The specificity of computer networks lies in the fact that they are distributed and operate by transmitting messages, which are described in the form of protocols. Therefore, the course deals with the fundamental concepts of computer networks used today: special attention is paid to the standard seven-layer protocol model and Internet protocols. Particular attention is paid to the development of network applications.
Topics	Computer Networks and the Internet - Introduction. Principles of Network Applications. The Web and HTTP. Electronic Mail in the Internet. Protocols of the Transport Layer Protocols of the Network Layer.

Name	Introduction to SQL Databases
Programme	bachelor
Semester	5
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	This course learns how to create relational databases, write SQL statements to extract information to satisfy business reporting requests, create entity relationship diagrams (ERDs) to design databases, and analyze table designs for excessive redundancy. This course ensures a common foundation for databases learners.
Topics	Principles of database architecture; concepts of database theory, purposes and main capabilities of DBMS. Principles of the architecture of information systems that use databases to store information. Principles of building single-table and multi-table queries/ Transact SQL programming. Basic concepts of Normal Form Theory.

Name	Theory and Methods of Relational Databases Design
Programme	bachelor
Semester	6
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students theoretical foundation for relational databases, Database normalization and functional dependency concepts. Relational algebra mainly provides a theoretical foundation for relational databases and SQL. Students use process of Normalization to minimize redundancy from a relation or set of relations.
Topics	Relational algebra. Functional and multi-valued dependencies. Anomalies in DBMS. Principles and main stages of database normalization. First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF), Boyce-Codd Normal Form (BCNF) and Fourth Normal Form (4NF). Decomposition. Lossless Decomposition.

Name	Web-programming Technology
Programme	bachelor
Semester	5
ECTS credits ,	4 ECTS, Credit, 2 2 hours lectures and 2 hours practical work per week

Exam / Credit	
Short description	The purpose of teaching the academic discipline is the formation of students' theoretical knowledge and practical skills aimed at the development of effective web applications using modern web development tools. Students should know: the basics of functioning of the World Wide Web; HTML hypertext markup language; content separation technology and design using cascading CSS style sheets; the basics of DHTML and the Document Object Model (DOM). Students should be able to: create static HTML pages and apply style sheets; create client scripts in javascript; have an idea of technologies based on the extensible XML markup language; apply the acquired knowledge to develop websites.
Topics	Site structure and web pages Block layout technology and types of site layouts XML, DTD, XSD, XSLT JavaScript conversion of types, functions, objects Using modules and classes to develop JavaScript programs SVG graphics, Canvas jQuery library

Name	Declarative Programming (Functional Languages)
Programme	bachelor
Semester	5
ECTS credits, Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course introduces the basics of fundamental topics underlying modern computer science such as computation theory, Turing machines and lambda-calculus. The lambda-calculus topic is introduced as the basis for the functional programming languages on the example of Scala language and described within theoretical practical and Scala coding assignments.
Topics	The fundamentals of algorithms and computations. Turing machine as a model of imperative computations approach. Basic of computation theory: definition of algorithm, computability, computably enumerable and computable sets, halting problem. Type-free lambda-calculus, church numerals, fixed-point combinator, Church–Rosser theorem. Introduction into Scala language, recursion and tail recursion, functions as first-class citizens, higher-order function, functional data structures.

Name	Methods of user interface design
Programme	bachelor
Semester	6
ECTS credits , Exam / Credit	3 ECTS, credit, 1 hour lectures and 2 hours practical work per week
Short description	The course is teaching students principles and methods of practical and usable GUI design. Psychological and neuro-physiological basics of human-computer interaction are considered. International standards on usability and metrics for usability evaluation are discussed. Students learn to build the requirements model, design the GUI architecture and apply these to their individual project. Team work is supposed. Peer-review of individual projects' usability is obligatory as well as the final public defense of teams' individual projects. CASE instruments, UML and Figma are used. The implementation of the project supposes free choice of technology stacks.
Topics	Steps to build the requirements model of domain area: purposes and actors; subject area dictionary, use cases, conceptual model, activities. UX and UI, levels of UX, concept of user interface architecture. Rules and principles of practical user interface. Neuro-physiological basics of human-computer interaction and their application during design. Coloristic. Concept of usability, international standards on usability, metrics of usability, methods of usability testing. Course project: 3 level MVC application with practical GUI. Usability report.

Name	Introduction to Parallel Processes Programming (C ++ and Java Languages)
Programme	bachelor
Semester	6
ECTS credits , Exam / Credit	4 ECTS, Exam, 2 hours lectures and 2 hours practical work per week
Short description	This course is an introduction to the basic concepts of concurrency programming and multithreaded programming. Students studying this course will learn how to design concurrency algorithms, develop multithreaded programs, distribute data between threads and avoid the problem of data races, and synchronize concurrency operations etc
Topics	Basic thread management operations. Methods of passing parameters to a thread Problems related to the state of data races. Mutexes and Deadlocks. Types of mutexes. Using conditional variables. Mechanism of future results. std::promise. Thread-safe abstract data types with locks and Non-locking thread-safe abstract data types Multithreading in Java. Life cycle of a thread

	Java thread pool.
--	-------------------

Name	Development of Compilers for Domain Specific Languages
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	3 ECTS, Credit, 2 hours lectures and 1 hour practical work per week
Short description	The course is intended for explaining the concept of Domain Specific Languages and training development techniques of compilers for such languages.
Topics	The concepts of Model-Driven and Domain Engineering. Semantic, Abstract and Concrete Syntax of a domain specification. Compiler Architecture Model. Design of a lexer, the compiler component for lexical analysis. Design of a parser, the compiler component for syntax analysis. Design of a translator, the compiler component for translation. Survey of tools for compiler design.

Name	Parallel and Distributed Computing
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	4 ECTS, exam, 2 hours lectures and 2 hours practical work per week
Short description	This course introduces the basic principles of distributed computing, highlighting common themes and techniques. In particular, we study some of the fundamental issues underlying the design of distributed systems.
Topics	Introduction into Parallel and Distributed Computing. Architectures of Distributed Computing. Processes into Distributed Computing. Communication into Distributed Computing. Coordination into Distributed Computing. Naming into Distributed Computing. Consistency and replication Fault tolerance

Name	Introduction to Artificial Intelligence
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, exam, 2 hours lectures and 2 hours practical work per week
Short description	A modern complex technical system is, as a rule, a system with intensive use of artificial intelligence. The specificity of artificial intelligence lies in

	the fact that they use special methods to solve problems that do not have a solution algorithm. Therefore, the course deals with the fundamental concepts of artificial intelligence used today: special attention is paid to the expert systems, fuzzy logics, genetics algorithms and neural network. Particular attention is paid to the development of AI applications.
Topics	The foundation of AI- Introduction. History and Applications AI. Intelligent agents. Problem Solving. Knowledge, reasoning and planning. Fuzzy logic. Genetic Algorithms Neural Networks

Mathematics Electives

Name	Computing Theory
Programme	bachelor
Semester	3
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course explores fundamental concepts in the theory of computation. Students analyze the limits of computability and gain insights into the mathematical foundations of computation, study the behavior of abstract machines and recursive functions.
Topics	Formal concept of algorithm and computable function Theoretical computing models: Turing machines, Unlimited registers machines, recursive functions Gödel enumerations of algorithms Undecidable and partially decidable problems Basic computational complexity theory

Name	Automata Theory
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course introduces students to the concept of a sequential computational system and mathematical models related to this concept. The notion of a state machine and its main kinds - acceptors Mealy's and Moore's transducers are the base for this introduction. Finite acceptors and their relationship with regular expressions are considered in detail. Algorithmic problems of regular languages are considered. Theoretical

	concepts are illustrated with associated programming constructions using Python programming language.
Topics	Concepts of a system and a model. The concept of a state machine. Mealy and Moore transducers. Representation of a computational process using an interaction between a finite Mealy transducer and an infinite Moore transducer. A finite state acceptor. What languages are recognised by finite state acceptors? Regular expressions and regular languages. Properties of regular languages. Brzozowski's derivative of a language. Differential calculus of regular expressions. Acceptor synthesis problem for a regular expression. Synthesis algorithm using differential calculus of regular expressions. Minimization problem for finite acceptors. Table filling and Hopcroft minimization algorithms. Application of regular expressions for automation of compiler design.

Name	Introduction to Numerical Methods
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, , Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students to the modern methods of approximate calculus. Such topics as machine arithmetic and floating-point computation, system of linear equations, data interpolation and approximation, numerical integration, solution of nonlinear equations, solution of ordinary differential equations, modelling of random numbers are considered. Students use Python language for practical work.
Topics	Machine arithmetic and floating-point computation. System of linear equations. Interpolation and approximation. Numerical solving of algebraic equation. Numerical integration Numerical solution of ordinary differential equations.

Name	Mathematical Basics of Computer Graphics
Programme	bachelor
Semester	5
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hour lectures and 2 hour practical work per week
Short	The course covers mathematical methods of computer graphics, namely:

description	geometric transformations on a plane and in space, construction of plane projections of spatial bodies, use of quaternion algebra in geometric transformations, use of spline curves and surfaces for construction of plane and spatial figures, algorithms for removing hidden lines and surfaces, mathematical methods of computer animation, mathematical foundations of fractal graphics.
Topics	Coordinate systems. Types of spaces. Movements on a plane and in space. Parallel projections. Central projections. Stereographic projection. Geometric algorithms of machine graphics. Curves on the plane and in space. Surfaces in space. Spline curves and surfaces. Mathematical foundations of animation. Fractals.

Name	Analytical Methods of Geometric Modelling
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, Credit, 2hour lectures and 2 hour practical work per week
Short description	The course teaches students to modern methods of modeling complex geometric objects on a computer. Issues of interpolation, approximation, theory of splines and methods of their analytical description are considered. Modern algorithms for constructing explicit and parametric equations of composite surfaces as well as implicit equations of boundaries of three-dimensional bodies are discussed and implemented. Students use the Maple system and the Python language for practical work.
Topics	Interpolation and approximation by polynomials and splines Explicit and parametric equations of compound curves and surfaces Application of the R-function method for constructing boundaries of three-dimensional bodies Inverse problems of analytical geometry. Application of geometric modeling methods in scientific graphics

Name	Mathematical Methods of Image Processing
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	4 ECTS, , Credit, 2hour lectures and 2 hour practical work per week
Short description	The course teaches students to some modern image processing techniques. Ways of presenting digital monochrome and color images, as well as mathematical methods of their processing and enhancement are

	considered. Algebraic methods of processing monochrome images, spatial filtering, two-dimensional discrete Fourier transform, filtering in the frequency domain, image enhancement methods in the presence of noise and the main methods of color image processing are discussed and implemented. Students use Matlab system and Python programming language for practical work.
Topics	Algebraic methods of processing monochrome images. Spatial filtering of images. Two-dimensional discrete Fourier transform. Image filtering in the frequency domain. Image enhancement methods in the presence of noise. Methods of color image processing.

Computer Science Electives

Name	Introduction to programming with Python
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, credit, 4 hours practical work per week
Short description	The course is teaching students to use Python's data types and data structures in problem solving, to understand dynamic model of Python programming and compare it with static model in C programming.
Topics	Python's operators, operations, data input and output. Working with simple data types. Modifiable and unmodifiable structures: strings, tuples, lists. Comparison, iteration and effective use in problems solving. Working with dictionaries. Functions, nested functions, LEGB rule, lambdas, closures. Working with files. Exceptions handling.

Name	Introduction to Programming with .NET Platform
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, Credit, 4 hours practical work per week
Short description	The purpose of teaching the academic discipline is the formation of basic knowledge and skills in the field of visual and object-oriented programming technology on the NET platform. Students should know: theoretical basics of programming in the C# language; programming technology in visual and object-oriented programming environments; basic concepts of event-driven programming for Windows OS.

	Students should be able to: develop a user interface using the simplest control elements in a visual integrated designer environment; work in a visual object-oriented C# programming environment (creating a project, compiling, debugging and testing programs; development and use of objects); apply methods and technologies of structural and visual programming when solving professional information processing tasks in the C# programming language.
Topics	Creating a console program Creation of a program with a window interface Peculiarities of program code debugging in the tool environment

Name	Introduction to Programming with JavaScript
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, Credit, 4 hours practical work per week
Short description	With the general introduction of JavaScript in the first course, this course will focus on JavaScript libraries, specifically jQuery. The scripts will be geared toward Document Object Model (DOM) manipulation. Learners will describe the top JavaScript libraries and differentiate which would work best to implement and accelerate development of web page animation.
Topics	This course introduces the programming language JavaScript and shows the websites that include the type of interactions students will eventually be able to develop. Learners will understand the importance of how JavaScript was developed and why such history impacts the way JavaScript is currently written and in future releases. Learners will write their first scripts, have their HTML and CSS skills assessed, create variables and arrays and assign values to them. If student's skills are lacking, resources and recommendations are provided to improve these skills. There is ample opportunity for students to practice these first, core skills.

Name	Version Control with Git
Programme	bachelor
Semester	2
ECTS credits , Exam / Credit	4 ECTS, Credit, 4 hours practical work per week
Short description	In this course, students learn how to track different versions of your code and configuration files using a popular version control system (VCS) called Git.
Topics	Throughout this course, students gain hands-on knowledge of Git's core features to understand how and why they are used in organizations,

	covering both basic and more advanced features such as branching and merging. Covers how to use VCS to work with others through remote repositories, such as those provided by GitHub.
--	--

Name	Programming with Python
Programme	bachelor
Semester	3
ECTS credits , Exam / Credit	4 ECTS, , Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course teaches students to the Python programming language and its capabilities for performing scientific calculations. Introduction to the main structure of the Python programming language and basic scientific packages numpy, matplotlib, sympy, scipy is given.
Topics	Data types and control operators Main functions from standard libraries Creating functions, modules and packages Working with arrays, vectors and matrices in the numpy package The matplotlib scientific graphics package. The SymPy symbolic mathematics package The main capabilities of the scientific computing package SciPy

Name	Algorithms of Computational Geometry
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students to working with algorithms of computational geometry. Classic algorithms of building of convex hull, line intersection search, polygon triangulation, point location task, Voronoi diagrams building, Delaunay triangulation, geometric search and query, geometric data structures, computational geometry in computer graphics, visibility and hidden surface removal, geometric optimization problems are considered in the course. Students use Python or C++ language for practical work.
Topics	Convex Hull: This algorithm finds the smallest convex polygon that contains a set of points. Line Intersection: It determines whether two lines or line segments intersect and calculates the intersection point if they do. Polygon Triangulation: This algorithm decomposes a polygon into a set of triangles. Point Location: It determines the position of a query point relative to a given geometric structure, such as a polygon or a line arrangement. Voronoi Diagrams: These diagrams partition a space into regions based on

	the distance to a set of input points. Delaunay Triangulation: It constructs a triangulation of a set of points, maximizing the minimum angle of all triangles.
--	---

Name	Programming with C#
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS,Credit, 2 hours lectures and 2 hours practical work per week
Short description	The purpose of teaching the educational discipline is the formation of a system of concepts, knowledge, abilities and skills in the field of modern programming in the C# language, which includes methods of design, analysis and creation of software products based on the use of the capabilities of the NET platform. To teach students to use modern integrated tool environments designed for the development of programs in an interactive mode. Lay the foundations for the next courses dedicated to the creation of modern information systems. Students should know: basic types and basic operations; basics of software access to relational data; features of structures and enumerations as significant types; principles of managed code in .Net; syntax for defining methods and their parameters. Students should be able to: apply the language of integrated queries (LINQ); create and apply iterators in data collection processing tasks; use a procedural paradigm to solve problems; use different data structures (stack, queue, dictionary, list) to solve practical problems; apply conditional operators and loop operators; design programs based on Windows Forms; solve problems using arrays.
Topics	Graphics Device Interface MS SQL Server .NET 6 Execution of SQL commands DataSet SQL LocalDB LINQ

Name	Information Theory and Coding
Programme	bachelor
Semester	4
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is teaching students to fundamental theoretical knowledge in areas of application of the most effective coding methods that allow the

	transmission of a certain amount of information over a communication channel using a minimum number of symbols, both in the absence and presence of interference.
Topics	Information Theory. Entropy Information Theory. Information compression. Haffman code. Coding Theory. Error-correcting coding Cycling code. Hamming code.

Name	Methods and Technologies of Big Data Processing
Programme	bachelor
Semester	5
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 4 hours practical work per week
Short description	The main tasks of studying the academic discipline are: mastering the basic concepts in the field of big data, getting acquainted with the principles of organizing big data stores, with basic algorithms for saving and searching in a big data store, getting acquainted with the list of tasks related to the organization of work with big data, and methods solving such problems, forming skills in the implementation of big data storage, organization of work processes with the storage, typical analysis of big data.
Topics	1. Introduction to Big Data: Basic concepts of Big Data; Characteristics of Big Data; Examples of Big Data problems; Modern Big Data technologies Containerization. Docker 3. MapReduce paradigm: MapReduce in distributed computing; Aspects of MapReduce operation; Basic stages of MapReduce; Basic patterns of the MapReduce paradigm; Application examples; 4. Hadoop distributed computing platform: Hadoop cluster characteristics; Distributed file system – HDFS; Cluster resource manager – YARN; Hadoop cluster monitoring tools; Running computing tasks on a Hadoop cluster (locally); Hadoop cluster installation and configuration features (locally) 5. SQL: Basics of SQL; Data Definition Language; Data Manipulation Language; Additional SQL topics (window functions, nested queries, joins) Orchestration of pipelines using Airflow (concept, UI and local customization, DAGs) Clouds 8. Familiarity with Apache Spark: Spark architecture; Comparison of Apache Spark and Apache Hadoop; Preparation for working with Spark APIs (introduction or repetition of SQL, overview acquaintance with Scala); Overview of Spark APIs (RDDs, DataFrames, DataSets); Spark monitoring tools

	9. Distributed data storage environments: Introduction to Apache Cassandra; Introduction to Vertica/Teradata; Introduction to Snowflake
--	---

Name	Object-oriented Programming with C#
Programme	Bachelor
Semester	6
ECTS credits , Exam / Credit	4 ECTS,Credit, 2 hours lectures and 2 hours practical work per week
Short description	The purpose of teaching the academic discipline is to form students' ideas about the direction of development of industrial programming platforms for building large corporate software systems and about the component-oriented programming paradigm. Students should know: OOP technologies and methods of organizing interaction in software systems, ensuring information security, extensibility and support of software solutions provided by the .Net platform. Students should be able to: make a comparative analysis and justify the choice of technologies and platform templates for building a software system; determine the conditions and limitations of applicability of various technologies and methods of the .Net platform depending on the specifics of the software project; use OOP technologies and patterns of the .Net platform to build software systems.
Topics	NET Framework Basic concepts of the object-oriented approach: objects, classes and methods and their implementation in C# Windows Forms, WPF Patterns: Abstract Factory, Prototype, Builder, Factory Method, Interpreter, Iterator, Command, Observer, Visitor, Mediator Principles of SOLID

Name	Advanced Scala Programming
Programme	bachelor
Semester	6
ECTS credits, Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course introduces the basics of simply typed lambda calculus and relations to the Scala type system, monadic data structures and concurrency with pipelines based monadic primitives.
Topics	Introduction to simple typed lambda-calculus by Church and Curry.

	Curry–Howard correspondence. Basics of monadic data structures. Type-classes in Scala: functors, groups, monoids. Monads in Scala. Introduction to cats library in Scala.
--	--

Name	Color and Light Models in Computer Graphics
Programme	bachelor
Semester	6
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course studies the modeling of geometric and physical properties of computer graphics. The main principles and methods of computer graphics, the main color models and algorithms of computer graphics are considered, the methods of pixel, vector and fractal graphics are considered, the principles and methods of creating visual scenes on a computer monitor are studied, methods of visual information processing.
Topics	Physical and biological foundations of computer graphics. Color models and laws of light propagation. Color models RGB, CMYK, HSB, HSV, HSL, CIE XYZ, Lab. Pixel graphics. Principles of vector graphics. Tools for creating vector images. The structure of vector illustration. Modeling of physical properties of materials: color, transparency, shininess, refractive index, specular reflection, diffuse lighting. Presentation and means of video information processing.

Name	Modern Java technologies
Programme	bachelor
Semester	6
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The aim of this course is learning modern frameworks in Java. Working with such technologies, students are getting knowledge related to back-end and front-end development. The course suggests learning Servlets API, JDBC, Hibernate, Spring MVC, IoC, Spring Security, Spring Boot and Spring Data JPA. According to this course, students will be able to develop scalable web applications following SOLID and modern software development principles.
Topics	Introduction to the course SQL queries recollection Maven

	JDBC Servlets/JSP JPA/Hibernate Spring (IoC, MVC, Security) Spring Boot, Spring Data JPA
--	--

Name	Introduction to Web-development with Java
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	This course is an introduction to the basic concepts of concurrency programming and multithreaded programming. Students studying this course will learn how to design concurrency algorithms, develop multithreaded programs, distribute data between threads and avoid the problem of data races, and synchronize concurrency operations etc
Topics	Basic thread management operations. Methods of passing parameters to a thread Problems related to the state of data races. Mutexes and Deadlocks. Types of mutexes. Using conditional variables. Mechanism of future results. std::promise. Thread-safe abstract data types with locks and Non-locking thread-safe abstract data types Multithreading in Java. Life cycle of a thread Java thread pool.

Name	IoT Systems Construction for Enterprises and Organizations
Programme	bachelor
Semester	7
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is intended for students studying the technologies of the Internet of things. To work with which, quite a variety of skills and knowledge are required, for example, programming languages and software packages for modeling IoT systems. The course contains a practical component on modeling the behavior of built system and its adaptation to different conditions.
Topics	Concept of IoT. Type of network topology, to implement in systems. Useful modules in C# for development of IoT. Modeling of fully func. System ic Cisco Pocket Tracer. Parsing and analyzing existing systems.

	Modeling and debugging own system. Practical aspects in implementation
--	---

Name	Categories Theory in Computer Science
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The course is designed to introduce the basic concepts of category theory and their application in computer science. These concepts are illustrated by corresponding software constructs, which are mainly related to functional programming.
Topics	Category Theory vs Set Theory as the background of Computer Science. Notions of a category, functors, and natural transformation. Constructions in categories. Universality. Universal arrows and objects. Yoneda's Lemma. Isomorphismic and equivalent categories. Adjunctions. Monads and comonads. Algebras and coalgebras. Initial and final semantics.

Name	Web-programming with Java
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hour lectures and 2 hour practical work per week
Short description	This course provides students with knowledge of the main Java frameworks used to develop web applications. Students learn the principles of building web applications using frameworks, the most successful technology stacks, and combining frameworks.
Topics	Using ORMs. Overview of existing ORMs. Hibernate: Steps of creating a project Annotations in Hibernate. HQL query language Spring Framework. Spring Web MVC, Spring Boot Spring Boot: Steps of development a project The REST protocol. Concept of RESTful web service

Name	Methods of Computer Networks Management
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, 2 hours lectures and 2 hours practical work per week

Short description	The course is teaching students to basic methods of Computer Networks Management, to work with Packet Cisco tracer, configure routers and works with different type of protocols.
Topics	Static routing. Protocol OSPF. Protocol BGP. VPN networks.

Name	Introduction to Machine Learning
Programme	bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	Introduction to Machine Learning is a foundational bachelor-level course that provides students with a comprehensive overview of the principles and techniques used in machine learning. Students will learn the fundamentals of supervised and unsupervised learning algorithms, data preprocessing, model evaluation, and feature selection. Through practical assignments and projects, students will gain hands-on experience in applying machine learning algorithms to real-world problems.
Topics	1. Introduction to Machine Learning: - Overview of machine learning concepts and terminology - Historical developments and applications of machine learning - Importance and impact of machine learning in various fields 2. Supervised Learning: - Regression: Linear regression, polynomial regression, and regularization - Classification: Logistic regression, decision trees, and random forests - Evaluation metrics: Accuracy, precision, recall, F1-score, and ROC curves 3. Unsupervised Learning: - Clustering: K-means, hierarchical clustering, and DBSCAN - Dimensionality reduction: Principal Component Analysis (PCA) and t-SNE - Anomaly detection: Outlier detection techniques 4. Model Evaluation and Validation: - Training, validation, and test sets - Cross-validation techniques: k-fold cross-validation and stratified sampling - Performance measures: Confusion matrix, precision-recall curve, and ROC analysis 5. Feature Engineering and Selection: - Feature extraction and transformation techniques - Feature scaling and normalization - Feature selection methods: Filter methods, wrapper methods, and embedded methods

	6. Advanced Topics in Machine Learning: - Ensemble learning: Bagging, boosting, and stacking - Deep learning: Neural networks, convolutional neural networks, and recurrent neural networks - Reinforcement learning: Markov decision processes, Q-learning, and policy gradients
--	---

Name	Multithread Programming and Web-development with .NET platform
Programme	Bachelor
Semester	8
ECTS credits , Exam / Credit	4 ECTS, Credit, 2 hours lectures and 2 hours practical work per week
Short description	The focus of this course is on learning object-oriented programming in C# for developing multi-threaded programs and web applications. This allows students to be introduced to the basic techniques of modern software development using an object-oriented approach and relevant practical skills. The goal of mastering the discipline is to provide students with knowledge and relevant practical skills regarding the basic methods of modern software development using an object-oriented approach in the development of multi-threaded programs and web applications. Students should know: C# capabilities related to object-oriented programming; capabilities of the .NET platform for creating web applications, know design patterns: architectural, behavioral; concepts and principles of multithreading and parallel programming in C#. Students should be able to: use ASP.NET Core design patterns in C#; create multi-threaded applications and use thread synchronization mechanisms; create ASP.NET Core MVC web applications.
Topics	Software implementation of multithreaded calculations on the .NET platform Introduction to ASP.NET Core ASP.NET Core MVC framework