Transilvania University of Brasov, Romania
Study program: Computer Science

Faculty: Mathematics and Computer Science
Study period: 3 years (bachelor)

1st Year – 1st Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Mathematical analysis	5	3	2	0	0

Course description (Syllabus): Relations (properties of a relation, equivalence relations, order relations, function as an example of a relation); Using axiomatic properties of the set of real number; Sequences and series (sequences and series of real numbers, sequences and series of functions); Determining the radius of convergence and interval of convergence of power series; Using the Taylor series expansions of function; First order and higher order partial derivatives of real-valued functions of several variables, the Schwarz theorem, the first order differential and the partial derivatives of a composite real function; Study extremes of the real differentiable functions of several variable; Constrained extrema, the method of Lagrange multiplier; Establish the nature and calculation of improper integrals; Calculation of integrals depending on a parameter; Calculation of line integral; Calculation of multiple integrals.

Course title	No. of credits	Number of hours per week			
		Course	seminar	laboratory	project
Basic algebra for computer science	5	2	2	0	0

Course description (Syllabus): Functions and relations, Binary operations, Free semigroups, Groups, Permutations, Rings and fields, Tropical geometry, Matrices, Vector spaces, Linear codes.

Course title	No. of credits	Number of hours per week			
		Course	seminar	laboratory	project
Fundamental algorithms	5	2	1	2	0

Course description (Syllabus): This course introduces fundamental concepts in algorithms, covering the building blocks of algorithm design. Topics include pseudocode, variable and constant usage, instructions, and visibility of variables (global/local) within an algorithm. The curriculum explores subprograms, recursion, and analysis of algorithm efficiency, including computing complexity across various scenarios. Specific algorithmic categories such as search and sort algorithms, divide and conquer algorithms, greedy algorithms, dynamic programming algorithms, and text processing algorithms are discussed. The course also involves a comparative analysis of implementations and programming methods. Additionally, it goes into intelligent algorithms, providing a foundational understanding of artificial intelligence concepts through research and case studies on selected issues. Practical components include problem-solving seminars on fundamental algorithm topics and hands-on laboratory sessions for the implementation of discussed algorithms.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Fundamentals of Programming	5	2	0	2	0

Course description (Syllabus): Getting familiar with the C language; Generic programming using macros; Understanding procedural programming; Pointers. Memory allocation and management; Developing function-oriented C application

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Mathematical and computational logic	5	2	1	0	0

Course description (Syllabus): Binary, decimal, octal, hexadecimal number systems; Conversions among bases; Addition, subtraction, multiplication and division of base b numbers, Representation of integer numbers (Sign and magnitude, One's complement, Two's complement); Addition, subtraction and multiplication of two's complement numbers; Fixed point numbers; Floating Point Numbers. The IEEE 754 Floating Point Standard (Formats and Rounding), Properties of Boolean Algebra; Truth Table; Boolean Functions. The disjunctive normal form and conjunctive normal form; Minimization of Boolean functions: Veitch-Karnaugh Maps, Quine McCluskey's method; Combinatorial circuits; The Algebra of GF (2). The operational and function domains; Reed-Müller expansions; Generalized Reed-Müller expansions.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Scientific and Professional Writing and Communication	2	1	1	0	0

Course description (Syllabus): This course offers a practical introduction to LaTeX for creating well-structured scientific and technical documents. Students learn the logical organization of LaTeX documents, text formatting, and the typesetting of mathematical formulas. The course covers essential packages for writing, the use of colors and graphics, and tools for graphing and presenting algorithms. Students also work with the beamer document class to create professional presentations and gain experience in preparing a complete scientific paper using LaTeX.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
English language 1 German language 1 French language 1 Spanish language 1	2	1	1	0	0

Course description (Syllabus): The English/German/French/Spanish language course attempts at revising, consolidating and improving grammar points at advanced level. It also provides opportunities for students to practice grammar structures and express themselves during the seminars. To this end, each lecture presents a theoretical issue which synthesizes essential information from outstanding books in English/German/French/Spanish morphology, followed by a wide range of exercises which will be approached during the following seminar. The exercises are designed in such a way so that students can solve them both individually and in pairs, during the seminars. The aim of this course is to provide students with comprehensive grammar structures linked to the following issues: problem verbs, modal verbs, passive voice, determiners and pronouns and relative clauses.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Physical education and sport	1	0	1	0	0

Course description (Syllabus): This course introduces fundamental principles of physical fitness, health, and various sports, promoting active lifestyles and teamwork.

1st Year – 2nd Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Linear algebra, analytical and differential geometry	5	2	2	0	0

Course description (Syllabus): Basic notions of linear algebra (vectors, calculus with vectors, dependence and independence linearity, bases, dimensions, linear applications, bilinear forms, and quadratic forms, scalar products and other products); Matrix representation from linear algebra (of vectors, linear applications, bilinear forms and quadratic forms, of vectors products); Basic notions of analytical geometry (point, line, conic, plane, quadric and its equations, frame, relative positions, angles, distances); To apply techniques from linear algebra in analytical geometry, the knowledge of basic elements from differential geometry of curves and surfaces; Explain the necessity for use specific techniques of linear algebra and analytical geometry in mathematics, physics, technique and informatics; Explain the use of specific techniques of linear algebra and analytical geometry (in particular and in details); Matrix interpreting the abstract definitions from linear algebra; Mathematics significations interpreting of some calculated elements; Use theoretical knowledge for problem solving; Interpreting algebraic and geometric some notions related to informatics.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Computer systems architecture	5	2	1	1	0

Course description (Syllabus): This course provides an in-depth understanding of the fundamental structure and internal operation of computer systems. You'll explore essential hardware components such as the Central Processing Unit (CPU), memory units, and Input/Output (I/O) systems, including peripheral devices. The curriculum delves into the digital-logic level representation of data, logic circuits, and registers, illustrating how information is internally processed. Key topics also include the detailed organization of the CPU, instruction sets, addressing modes, and the execution of programs. Additionally, you'll be introduced to assembly language programming and advanced architectural concepts like multiprocessing, multi-computer systems, and parallel, concurrent, and distributed computing.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Object oriented programming	6	2	0	2	0

Course description (Syllabus): This course offers an in-depth exploration of advanced C++ features and the fundamental principles of Object-Oriented Programming (OOP). You will master dynamic memory management, templates, exception handling, operator overloading, and lambda expressions. The core of the course focuses on OOP concepts such as class declaration, constructors, destructors, inheritance (including multiple and virtual inheritance), polymorphism through virtual functions, and the use of friend and static members. Additionally, you will learn about C++ streams for efficient input/output operations, file processing, and C++ string manipulation.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Data structures	6	2	0	2	0

Course description (Syllabus): The course presents a series of data structures used to efficiently store and manipulate data in computer programs. It begins with simple structures such as stacks, queues, linked lists, and hash tables. It then introduces several types of trees, with emphasis on binary search trees, balanced trees, and heaps. Finally, more advanced data structures, including B-trees, disjoint sets, and quadtrees, are discussed. The data structures are presented in terms of their applicability, the main operations they support, and the associated time complexity. In addition, the corresponding structures and objects from the C++ STL library are introduced. During the laboratory sessions, students implement these data structures and use them in the context of problem solving.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Operating systems	5	2	0	2	0

Course description (Syllabus): This course provides an in-depth understanding of the fundamental principles and intricate workings of operating systems. Students will explore OS goals, architecture, and essential resource management across processes, memory, I/O, and storage, alongside concepts like security, protection, virtualization,

and distributed systems. The curriculum covers OS structures, services, user interfaces, system calls, and the boot process, with a significant focus on file systems, including their concepts, access methods, directory structures, allocation strategies (contiguous, linked, indexed), and recovery mechanisms. Process management is examined in detail, encompassing processes, CPU scheduling algorithms (FCFS, SJF, Round Robin, Priority, Multilevel Queue), context switching, inter-process communication (IPC), threads, concurrency, multi-core programming, and remote procedure calls (RPC). The course thoroughly addresses process synchronization, covering race conditions, critical sections, mutexes, semaphores, monitors, classic synchronization problems (e.g., bounded buffer, readers-writers, dining philosophers), and robust strategies for deadlock prevention, detection, and recovery.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
English language 2 German language 2 French 2 Spanish 2	2	1	1	0	0

Course description (Syllabus): The main objective of English/German/French/Spanish language seminars from the second semester is that of consolidating and improving the English/German/French/Spanish language knowledge acquired by students up to that point. This time the focus is no longer grammar but students' ability to express themselves as fluently and as accurately as possible by means of using a wide range of vocabulary items. To this end special material providing them with texts and exercises covering various artistic, cultural, political and social issues will be used. Apart from the emphasis placed on developing students' speaking skills, the seminars will also focus on students' ability to communicate in writing. They will be taught to develop and organize their ideas logically and coherently, while at the same time using the language correctly.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Physical education and sport	1	0	1	0	0

Course description (Syllabus): This course introduces fundamental principles of physical fitness, health, and various sports, promoting active lifestyles and teamwork.

2nd Year - 1st Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Graph algorithms	5	2	0	2	0

Course description (Syllabus): This course offers an in-depth exploration of graph theory and its essential algorithms, providing students with the tools to model and solve complex problems. The curriculum begins with introductory concepts in graph theory, laying the foundation for understanding various graph structures and properties. Students will then delve into graph traversal techniques, including partial and full traversals, as well as topological sorting. The course covers the identification of connected components and strongly connected components and explores the fundamental concepts of cycles and trees. A significant portion is dedicated to algorithms for finding minimum spanning trees. Further advanced topics include arborescence, algorithms for computing distances and shortest paths, and a thorough study of network flows. By the end of this course, students will possess a strong theoretical and practical understanding of graph algorithms, enabling them to apply these powerful techniques across various computational domains.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Formal languages and automata theory	5	2	1	1	0

Course description (Syllabus): Automata theory stands at the core of theoretical computer science, encompassing important concepts essential for various applications. This course explores the theoretical formalization and automata, providing insights into their role in programming language compilation, text editor construction, and network modeling. Participants will develop skills in working with formal elements intrinsic to theoretical computer science. Key concepts covered include Markov systems, grammars, Chomsky classification, automata, push-down automata, regular expressions, and various types of analysis within the compilation flow.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	Project
Programming environments and tools	5	2	0	2	0

Course description (Syllabus): This course begins with introductory elements, exploring the fundamental question of why object-oriented programming is significant. It explores the key concepts and principles of object-oriented programming, focusing on specifying and implementing a class with a Java perspective. The course covers inheritance and polymorphism in object-oriented programming, providing insights from the Java standpoint. Additionally, the module addresses structured exception handling in object-oriented programming, the intersection of object orientation and generic programming in a Java context, and explores object-oriented flows and object serialization, offering a comprehensive understanding of these essential aspects.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	Project
Databases	5	2	0	2	0

Course description (Syllabus): This course offers a comprehensive exploration of database systems, covering the fundamental distinction between logical and physical data organization. Students will master data modeling techniques, including logical and physical database design, and understand the ANSI/X3/SPARC model. A core focus is placed on the Entity-Relationship (ER) model and practical design rules, guiding through the main steps of logical database design. The curriculum also examines various operating modes and their representation. Practical application is emphasized through a case study on property association management using SQL, and the course concludes with insights into developing systems with databases. By the end, students will have a solid theoretical and practical foundation in database design, implementation, and management.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	Project
Artificial intelligence	5	2	0	2	0

Course description (Syllabus): This course provides a practical and theoretical introduction to Artificial Intelligence, with emphasis on implementation using Python. Students will begin with an overview of the AI field, understanding its core concepts and applications. The curriculum then delves into fundamental machine learning algorithms, starting with Linear Regression and Logistic Regression, essential for predictive modeling. We will explore the building blocks of neural networks, covering the Linear Perceptron and advancing to Multilayer Perceptron. Further topics include specialized network architectures such as Radial Basis Function Networks and Adaptive Resonance Models. The course concludes with the study of Convolutional Neural Networks (CNN) and of evolutionary models. By the end, students will understand key AI algorithms and know how to implement them using Python.

Course title	No. of credits	Number of hours per week			
		course	seminar	Laboratory	Project
Modern C++ managing networking problems	5	2	0	2	0

Course description (Syllabus): This course is centered around tracing the evolution of the C++ language. By mastering

the introduced concepts, participants will gain the ability to craft modern C++ code characterized by improved safety, speed, and simplicity. The C++ standard has undergone a continuous evolution with the introduction of C++11, C++14, C++17, and C++20. These updates facilitate writing more concise code, eliminating memory leaks, and enhancing overall runtime performance. Additionally, certain modules of the course focus on providing a C++ perspective for implementing practical, project-oriented networking components in hands-on labs.

Course title	No. of credits	Number of hours per week			
		course	seminar	Laboratory	Project
Physical education and sport	2	0	2	0	0

Course description (Syllabus): This course introduces fundamental principles of physical fitness, health, and various sports, promoting active lifestyles and teamwork.

2nd Year – 2nd Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	Laboratory	project
Automata, calculability and complexity	5	2	1	1	0

Course description (Syllabus): This course introduces the fundamental concepts and principles of computer science, with an emphasis on the mathematical theories and formal models that underpin the discipline. Students will learn to interpret mathematical and computational models and to identify appropriate techniques for solving real-world problems. The course also explores the use of simulation to analyze model behavior and evaluate system performance. Additionally, students will learn to incorporate formal methods into applications across various domains, with particular attention to accurately describing programming paradigms and language-specific mechanisms, as well as distinguishing between semantic and syntactic aspects. Building on these concepts, learners will develop source code and unit-test components in a familiar programming language, following given design specifications. The course further covers application testing based on structured test plans, as well as the development of program units and the preparation of accompanying technical documentation.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Advanced methods of programming	5	2	0	2	0

Course description (Syllabus): This course introduces visual programming with a focus on the .NET Framework and .NET Core. Students will cover fundamental concepts like data types, arrays, strings, instructions, classes, and namespaces. A core component is Object-Oriented Programming (OOP) in C#, alongside advanced features such as delegates, events, structs, collections, and generic classes. The curriculum also includes exception handling, attributes, ADO.NET, LINQ to objects, threads, and working with data streams. Students will be able to develop modern applications using the .NET ecosystem.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Computer networks	5	2	0	2	0

Course description (Syllabus): This course introduces the fundamental concepts of computer networks and provides an overview of current network technologies. Students acquire the knowledge necessary for distributed programming and develop practical skills in network administration. The course covers the use of Windows commands for network management, understanding IP and MAC addresses, and implementing various network algorithms. Additionally, students learn to design and configure local networks and interconnected networks, gaining hands-on experience in building and managing networked systems.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Numerical computation	5	2	0	2	0

Course description (Syllabus): This course introduces core numerical methods for solving linear and nonlinear mathematical problems. Topics include matrix types and transformations, LU factorization, iterative methods such as Jacobi and Gauss–Seidel, and techniques for computing determinants and matrix inverses. Students also study Newton's method and its variants for nonlinear equations and systems, as well as basic numerical integration using the trapezoidal and Simpson rules.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Digital image processing	5	2	0	2	0

Course description (Syllabus): This course introduces the fundamentals of digital image processing, exploring essential stages from data representation and visual enhancement to advanced segmentation and filtering techniques. It covers point operations, geometric transformations, spatial and frequency domain filtering, morphological operations, and methods for edge and texture detection. The objective is to provide a solid understanding and practical skills for manipulating and analyzing digital images.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Digital technologies for IoT	5	2	0	2	0

Course description (Syllabus): This course introduces the Internet of Things (IoT) within the broader context of the Web of Things, Cyber-Physical Systems, Autonomous Distributed Systems, and the Internet of Everything. Students explore core IoT concepts, applications, benefits, limitations, and related security challenges. Topics include sensors, actuators, sensor networks, basic robotics elements, stigmergy, intelligent and reactive systems, as well as IoT system architectures and the programming technologies that support them. The course also covers integrating microcontroller boards into IoT solutions and understanding how real-world interactions shape autonomous and connected systems.

3rd Year – 1st Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Software engineering	5	2	0	2	0

Course description (Syllabus): This course introduces the fundamental concepts, principles, and techniques of software development, with a focus on modern software engineering practices. Students develop practical skills in using support tools and methodologies while gaining specific knowledge of Agile frameworks such as Scrum, Kanban, and Extreme Programming. Emphasis is placed on understanding how to apply Agile practices effectively in real software projects.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Human-computer interfaces	5	2	0	2	0

Course description (Syllabus): This course prepares students to design and develop team-based projects in the modern Internet of Things (IoT) context, integrating both hardware and software components for practical applications. Students learn the essential concepts and characteristics of the IoT field, gaining experience in designing and assembling hardware prototypes and implementing the corresponding software. The course also emphasizes recognizing and analyzing IoT-specific challenges and formulating well-reasoned perspectives on current issues in the domain.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Development of web applications	5	2	0	2	0

Course description (Syllabus): This course introduces the structure of modern websites and the functioning of contemporary computer systems, enabling future computer scientists to use computing resources effectively. Students learn to prepare and integrate core web content, including text, graphics, animations, and audio, into functional site prototypes, and to perform online testing of their applications. The course also covers the integration of supporting technologies, databases, and e-commerce systems, as well as essential maintenance practices such as updating content, monitoring site performance, and adapting or extending site structure.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Professional practice	5	0	0	0	8

Course description (Syllabus): This course is designed to provide students with essential practical professional experience. It encompasses various forms of hands-on engagement, including internships within external companies or institutions, practical work conducted in university research laboratories, or active participation in faculty-led projects. The overall coordination and evaluation of these practical activities are managed by a designated practice coordinator for each respective study program.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Developing computer games in Unity with C#	5	2	0	2	0

Course description (Syllabus): The aim of the course is the study of programming concepts, which are the basis3D computer games; Students will acquire the following theoretical and practical skills aftergoing through the didactic activities: They will deepen the fundamental notions necessary for the development of a 3D engine inC# for games, platform independent (desktop or mobile), e.g.: Scripting, textures, Prefabs, Layers, Physics, Constraints. Meshes; Students will develop a 3D object rendering engine using Unity in C#, which will be able to run platform independent; Students will understand how the camera works and user interaction using the keyboard and mouse; Students will write vertex and fragment shaders that they will run directly on graphics card.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Front-end application architecture using Angular	5	2	0	2	0

Course description (Syllabus): This course offers a practical introduction to modern web development using Angular. Students learn the framework's core concepts, including components, modules, services, and communication mechanisms. The course covers directives, pipes, routing, guards, and navigation, as well as reactive programming with RxJS. Additionally, students explore unit testing and best practices for building robust, maintainable Angular applications.

3rd Year – 2nd Semester

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
IT project management	5	2	0	2	0

Course description (Syllabus): This course presents the fundamental concepts, technologies, and best practices for modern IT project management, with a focus on effectively managing software development projects. Students develop skills in organizing, estimating, and monitoring the progress of software projects, while also enhancing their ability to work collaboratively in development teams.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Parallel, concurrent and distributed programming	5	2	0	2	0

Course description (Syllabus): This course introduces parallel programming for distributed and GPU-based systems. Students learn the key components of programming languages and models used in these environments, with a focus on MPI for distributed architectures and CUDA for GPU computing. The course also covers fundamental parallel algorithms, providing the skills needed to design and implement efficient parallel solutions.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Probability and statistics	5	2	1	1	0

Course description (Syllabus): This course provides fundamental knowledge of probability and mathematical statistics and demonstrates their use in solving practical problems. Some of the most relevant topics include field of events and operations with events, classical and axiomatic definitions of probability, conditional probability, independent events, and key results such as the total probability and Bayes' formulas. Students study major probabilistic schemes, including hypergeometric, Poisson, binomial, multinomial, and Pascal distributions, and learn about random variables, discrete probability laws, and the distribution functions associated with discrete random variables.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Practical coordination for bachelor thesis	5	0	0	0	6

Course description (Syllabus): This course focuses on the supervised process of developing a bachelor thesis. It involves a designated coordinator who oversees the student's entire thesis elaboration journey, providing guidance and support. The final grade for this course is assigned by the individual coordinator, based on the student's engagement, progress, and the quality of work demonstrated throughout the thesis development.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Designing mobile applications	5	2	0	2	0

Course description (Syllabus): This course introduces the core components of programming languages for mobile application development and their application in designing and implementing projects. Students learn to work with user interface components, local and web databases, graphics, and animation, while exploring effective mobile application design methods. Emphasis is placed on applying these concepts to create functional and well-structured mobile applications.

Course title	No. of credits	Number of hours per week			
		course	seminar	laboratory	project
Introduction to quantic computation	5	2	0	2	0

Course description (Syllabus): This course provides an introduction to the foundations of quantum computing. It begins with a review of key concepts from linear algebra and mathematical statistics, followed by an overview of quantum mechanics relevant to computation. Students are introduced to fundamental quantum algorithms, quantum hardware, and the principles of quantum communication. The course also covers quantum networks, including quantum key distribution (QKD).