COURSE SYLLABUS

University	UNIVERSITY OF ORADEA
Faculty	FACULTY OF ENERGY ENGINEERING AND INDUSTRIAL MANAGEMENT
Study program*	INDUSTRIAL ECONOMICS ENGINEERING

I. Course Name: COMPUTERS PROGRAMMING AND PROGRAMMING LANGUAGES I

II. Course Details

No of hours/week						
Code	Semester	Credits	Lecture	Seminar	Laboratory	Project
IEMI-0162	1	6	2	-	2	-

III. Course coordinator (title, name, surname, e-mail):
Lecturer Vasile Moldovan, moldovan@uoradea.ro

IV. Course objectives

Application of basic knowledge, concepts, and methods regarding computer system architecture, programming languages, and techniques.
The course provides students with fundamental knowledge regarding the operation of a computer system and the general principles of structured programming.
The specific objectives addressed by the course focus on solving concrete practical problems involving programming elements, data structures, and algorithms, through the development of programs in a general-purpose and/or specific programming language.

V. Course content	**No. of hours**
V.1. Lecture (chapters/subchapters and paragraphs)	
C1. Presentation of computer system components. Memory structure; Data storage entities; Memory types. Generic data access methods: data/instruction addressing modes.	4
C2. Information representation in digital computers, numbering systems, alphanumeric codes, numeric codes.	2
C3. Algorithm representation methods. Flowcharts. Pseudocode.	4
C4. Theory of computer-aided problem solving. Problem-solving stages.	2
C5. Implementing algorithms in programming languages.	2
C6. Fundamental algorithms.	2
C7. Linear, branching, and cyclic algorithms (with known and unknown number of steps).	4
C8. Basic algorithms applied to character strings.	2
C9. Basic algorithms applied to arrays.	2
C10. Sorting and merging algorithms.	2
C11. Fast search algorithms.	2
V.2. Laboratory	
L1. Numeral systems: binary, octal, hexadecimal.	2
L2. Converting numbers from one numeral system to another.	2
L3. Binary addition. Representation of numbers in one's complement and two's complement.	2
L4. Applications with conditional structures.	2
L5. Applications with iterative (loop) structures.	2
L6. Applications with one-dimensional and two-dimensional arrays.	2
L7. Sorting algorithms.	2
L8. Search algorithms.	2
L9. Applications with subprograms (functions and procedures). Recursion.	2
L10. Examples of applications with files.	2
L11. Applications with characters and strings.	2
L12. Sorting and searching applications. Graph's representation.	2
L13. Verification test.	2
L14. Laboratory recovery.	2

VII. Grading criteria

Activities	Assessment	% of final grade
Exam	1 final verification test - Written exam (or multiple-choice test) on theoretical knowledge, including practical application. Evaluation Criteria: Correctness and completeness of knowledge; Logical coherence; Degree of assimilation of specialized terminology.	70 %
Laboratory	Ability to apply knowledge in practice; Criteria regarding attitudinal aspects: conscientiousness, interest in individual study. Periodic written assessments. Active participation in laboratory sessions.	20 % 10 %

VIII. Learning outcomes:

- Analyze the hardware and software components involved in program execution;
- Manipulate data representations and perform arithmetic operations in various base systems;
- Create logical models for computational problems using flowcharts and pseudocode;
- Write syntactically and logically correct code for linear, branching, and repetitive algorithms;
- Apply standard algorithms for sorting, searching, and inter-classing data structures (arrays, strings);
- Construct modular applications for specific tasks, such as large integer arithmetic and text manipulation

Course coordinator,
Lecturer Vasile MOLDOVAN, PhD., Eng.