MATHEMATICAL SOFTWARE
ANALYTIC SYLLABUS

Academic Year 2025-2026

Year of study II / Semester I

1. Information on academic programme

1.1. University	**„1 Decembrie 1918" University of Alba Iulia**
1.2. Faculty	**Faculty of Computer Sciences and Engineering**
1.3. Department	**Computer Science, Mathematics and Electronics**
1.4. Field of Study	**Computer Science**
1.5. Cycle of Study	**Undergraduate**
1.6. Academic programme / Qualification COR/ESCO	**Computer Science, COR 251201, 251204, 251203, ESCO-08: 2511/ Systems Analyst, 2512/ Software developers**

2. Information of Course Matter

2.1. **Course**	*Mathematical software*			2.2. **Code**		CSE206	
2.3. **Course Leader**			Full Prof. Ph.D., Dr. Habil., Nicoleta Breaz				
2.4. **Seminar Tutor**			Assist. PhD student, Andreea Cunțan				
2.5. Academic Year	**II**	2.6. Semester	**I**	2.7. Type of Evaluation (E – final exam/ CE - colloquy examination / CA -continuous assessment)	**CE**	2.8. Type of course (**C**– Compulsory, **Op** – optional, **F -** Facultative)	**Op**

3. Course Structure (Weekly number of hours)

3.1. Weekly number of hours	**4**	3.2. course	***2***	3.3. seminar, laboratory	***2***
3.4. Total number of hours in the curriculum	**56**	3.5. course	***28***	3.6. seminar, laboratory	***28***

Allocation of time:	Hours
a.Individual study of readers	***20***
b.Documentation (library)	***20***
c.Home assignments, Essays, Portfolios	***40***
d.Tutorials	***5***
e.Assessment (examinations)	***4***
f.Other academic activities (study visit, projects etc.)	***5***

3.7 Total number of hours for individual study	**80**
3.8. Total number of hours for academic activities	**70**
3.9 Total number of hours per semester	**150**
3.10 Number of ECTS	**6**

3. Prerequisites (*where applicable*)

4.1. curriculum-based	**-**
4.2. competence-based	-

4. **Requisites** (*where applicable*)

5.1. course-related	*The course is hosted in a room equipped with video projector and computers having installed Matlab/Octave/Python. The tutorials are at the students' disposal (in the library). The course materials will be uploaded also on Microsoft Teams (if it is available).* ***Note:*** *The students are strongly encouraged to attend the course, in order to gain knowledge for practical applications.*
5.2. seminar/laboratory-based	*The seminar is hosted in a laboratory equipped with video projector and computers having installed Matlab/Octave/Python. The tutorials are at the students' disposal (in the library). The homework for lab must be uploaded on Microsoft Teams (if it is available).* ***Note:*** *The attendance of the laboratory classes is compulsory, a student who doesn't attend all classes being not allowed at the exam (the attendance is checked by checking the uploading of all homework, in time). The missed classes can be recovered, by uploading the homework, until the end of the lab classes in the week 12.*

5. Specific competences to be acquired (chosen by the course leader from the programme general competences grid)

Professional competences	The course is focused on the development of skills required to use mathematical software and also to project some supplementary components, for a software, dedicated to solve new problems; the graduate will be able to solve various mathematical problems supposing large calculus, based on a software product. Aiming the development of these specific competences, the course assures the knowledge on mathematical software which contributes to the general professional competences given by the study program, regarding ***The use of computer tools in an interdisciplinary context*** **(C3).** These can be clearly described by the level descriptors related to: C3.3. The use of computer and mathematical models and tools to solve specific problems in the application field. C3.4. Data and model analysis C3.5. The development of software components of interdisciplinary projects.
Transversal competences	-

6. Course objectives (as per the programme specific competences grid)

7.1 General objectives of the course	The general aim related to this course consists in getting knowledge which allows to initiate students in the use of mathematical software products, applied in different problems with mathematical component, arising in various fields of science and technique and in general, in the ***use of computer tools in an interdisciplinary context.***
7.2 Specific objectives of the course	It is aiming the development of some specific competences to use mathematical software, thus the students will get the ability to use software product to solve problems that requires large and hard calculation and also to simplify the way how the results are returned. It is underlined the use of Matlab/Octave/Python mathematical functions, such that the students is in the end capable **to use computer and mathematical models and tools to solve specific problems in the application field, to analyses data and models, to develop software components of**

	interdisciplinary projects.

7. Course contents

8.1 Course (learning units)	Teaching methods	Remarks
I. Mathematical Software Toolboxes -general issues (2 hours) 1. The use of specific software in the solving of mathematical problems 2. Types of mathematical software **(Objectives: learning basics for using of the computer and mathematical models and tools to solve specific problems in the application field)**	***Lecture, discussion***	2 hours onsite Minimal lectures: 2 (see the list)
II. Introduction in MATLAB/Octave/Python* (12 hours) 1. About work sessions (1 hour) 2. Constants, variables, predefined functions, arithmetical, logical and relational operators (2 hours) 3. Instructions for reading, editing and assigning (1 hour) 4. Commands for 'script' m- file/py-file (2 hours) 5. Instructions for flow control, branching and efficiency evaluation (2 hours) 6. Functions (procedures) (2 hours) 7. Specific toolboxes/packages/libraries (2 hours) **(Objectives: learning basics for using of the computer and mathematical models and tools to solve specific problems in the application field)** ***Matlab can be replaced or completed with similar software Octave/Python/R**	***Lecture, discussion, exemplification in Matlab/Octave/Python****	12 hours onsite Minimal lectures: 2 (see the list)
III. Mathematical functions in MATLAB/Octave/Python* (14 hours) 1. Basic math functions (12 hours) 1.1. Functions for linear algebra and matriceal calculus (2 hours) 1.2. Functions for elementary math and trigonometric (2 hours) 1.3. Functions for data analysis (2 hours) 1.4. Functions for polynomial calculus (2 hours) 1.5. Functions for numerical methods (2 hours) 1.6. Functions for graphics (2 hours) 2. Applications in Matlab/Octave/Python* (2 hours) **(Objectives: to use computer and mathematical models and tools to solve specific problems in the application field, to analyses data and models, to develop software components of interdisciplinary projects.)** ***Matlab can be replaced or completed with similar software Octave/Python/R**	***Lecture, discussion, exemplification in Matlab/Octave/Python*, materials in digital format***	14 hours onsite Minimal lectures: 2 (see the list)
References **1. S. Attaway, MATLAB: A Practical Introduction to Programming and Problem Solving, 6th edition, Boston University,Butterworth-Heinemann, 2022** **2. N.Breaz, Mathematical software, Univ. "1 Decembrie 1918" din Alba Iulia, (electronic version)** **3. D. J. Higham, N. J. Higham, MATLAB Guide, 2nd edition, SIAM, 2005** **4. H.-H. Lee, Programming and Engineering Computing with MATLAB 2023, SDC Publications, 2023** **5. P. I. Katan, Matlab for beginners, a gentle approach, revised edition, Katan, 2023,** **6.P. Marchand, O. T. Holand – Graphics and GUI with MATLAB, 3rd edition, Barnes and Noble, 2003** **7. C. Moler – Numerical Computing in MATLAB, SIAM, 2005** **8. S. Nagar, Introduction to Octave - For Engineers and Scientists, Apress, 2018** **9. ***– Documentation for MathWorks Products, R2009a- http://www.mathworks.com/** **10. ***, Documentation for Octave, https://octave.org/** **11. ***, Documentation for Python: https://www.python.org/doc/**		
Seminars-laboratories		
1. Basics commends in Matlab/Octave/Python* (8 hours)* -Working with MATLAB/Octave/Python* session (2 hours) -Constants, variables, predefined functions, arithmetical, logical and relational operators (2 hours) -Instructions for reading, editing and assigning (2 hours) -Commends for 'script' m- file/py-file (2 hours) **(Objectives: learning basics for using of the computer and mathematical models and tools to solve specific problems in the application field)** ***Other similar software as Octave can be used in teaching and assessment as alternative or complementary to Matlab. Also, if time allows, other software as R/Python, can be also used.**	***Coordination and evaluation of computer based works***	8 hours onsite Minimal lectures: 2 (see the list)
2. Programming in Matlab/Octave/Python* (6 hours) -Flow control, branching and efficiency evaluation (3 hours) -Functions (procedures) (3 hours) **(Objectives: learning basics for using of the computer and mathematical models and tools to solve specific problems in the application field)** ***Matlab can be replaced or completed with similar software**	***Coordination and evaluation of computer based works, materials in digital format***	6 hours onsite Minimal lectures: 2 (see the list)

OCTAVE/Python		
3. The use of basic math functions in Matlab/Octave/Python *(14 hours) -Functions for linear algebra and matriceal calculus (2 hours) -Functions for elementary math and trigonometric (2 hours) -Functions for data analysis (2 hours) -Functions for polynomial calculus (2 hours) -Functions for numerical methods (2 hours) -Functions for graphics (4 hours) **(Objectives: to use computer and mathematical models and tools to solve specific problems in the application field)** ***Matlab can be replaced or completed with similar software Octave/Python**	***Coordination and evaluation of computer based works, materials in digital format***	14 hours onsite Minimal lectures: 2 (see the list)

References

1. S. Attaway, MATLAB: A Practical Introduction to Programming and Problem Solving, 6th edition, Boston University,Butterworth-Heinemann, 2022
2. N.Breaz, Mathematical software, Univ. "1 Decembrie 1918" din Alba Iulia, (electronic version)
3. D. J. Higham, N. J. Higham, MATLAB Guide, 2nd edition, SIAM, 2005
4. H.-H. Lee, Programming and Engineering Computing with MATLAB 2023, SDC Publications, 2023
5. P. I. Katan, Matlab for beginners, a gentle approach, revised edition, Katan, 2023,
6.P. Marchand, O. T. Holand – Graphics and GUI with MATLAB, 3rd edition, Barnes and Noble, 2003
7. C. Moler – Numerical Computing in MATLAB, SIAM, 2005
8. S. Nagar, Introduction to Octave - For Engineers and Scientists, Apress, 2018
9. *– Documentation for MathWorks Products, R2009a- http://www.mathworks.com/**
10. *, Documentation for Octave, https://octave.org/**
11. *, Documentation for Python: https://www.python.org/doc/**

1. Corroboration of course contents with the expectations of the epistemic community's significant representatives, professional associations and employers in the field of the academic programme

The skill's development regarding the use of a mathematical software and the stimulation for the premises to know how to project software products, adequate to different sciences, contribute to the complementarity's warranty required for a software programmer, this being capable to develop various software products having precise specifications (software having a mathematical component), without needing for mathematicians' help, the course answering in this way, to the necessity of the graduate to be adapted at various fields from the labor market, where specialists in computer science are needed.

2. Assessment

Activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	*Final evaluation:* *- correct interpretation of the results of the mathematical problems, solved by using a mathematical software, in the synthesis homework, will be assessed through a set of questions, during the final colloquy examination (questions based on applied theory within the synthesis homework);*	***Final colloquium examination** – presentation and questions (onsite) based on the synthesis homework.*	50%
10.5 Seminar/laboratory	*Continuous assessment* *- the students have to solve correctly, by using the presented mathematical software, all mathematical problems from their practical works required during lab classes*	***Continuous assessment:** During the lab classes, the assessment of practical skills in using a mathematical software will be done, by evaluating the portfolio containing all required practical homework.*	50%
10.6 Minimum performance standard:			
Correctly solving of some mathematical problems having a medium level of complexity, using mathematical software (for example, to solve a system of equations in Matlab/Octave/Python, to plot a basic mathematical function etc). **Note: *Please see also the alignment 5 (Requisites), related to compulsory attendance of the practical classes (mandatory requirement for exam admission is the uploading of all lab homework in term, with recovery in the week 12). Also, a student who doesn't attend the Final colloquy examination, can not get a final mark even the mark for continuous assessment is obtained. The assessment scale is from 1 to 10, 5 is minimum to pass the exam and in order to get ECTS, a student must get minimum 5 for each of the two tasks (synthesis homework and lab homework).***		

Submission date Course leader signature Seminar tutor signature

__25.09.2025__ _____ _____

Date of approval by Department members | Department director signature

_____ _____