SYLLABUS

ADVANCED METHODS FOR SOLVING MATHEMATICAL AND ALGORITHMIC PROBLEMS

University year 2026-2027

1. Information regarding the programme

1.1. Higher education institution	Babes Bolyai University
1.2. Faculty	Mathematics and Computer Science Faculty
1.3. Department	Computer Science Department
1.4. Field of study	Computer Science
1.5. Study cycle	Bachelor
1.6. Study programme/Qualification	Computer Science (English)
1.7. Form of education	

2. Information regarding the discipline

2.1. Name of the discipline	**ADVANCED METHODS FOR SOLVING COMPUTER SCIENCE PROBLEMS**					Discipline code	**MLE5199**
2.2. Course coordinator				Lect. Dr. Mircea Ioan-Gabriel			
2.3. Seminar coordinator				Lect. Dr. Mircea Ioan-Gabriel			
2.4. Year of study	1	2.5. Semester	2	2.6. Type of evaluation	E	2.7. Discipline regime	obligatory

3. Total estimated time (hours/semester of didactic activities)

3.1. Hours per week	**5**	of which: 3.2 course	**2**	3.3 seminar/laboratory/project	**3**
3.4. Total hours in the curriculum		of which: 3.5 course		3.6 seminar/laboratory/project	

Time allotment for individual study (ID) and self-study activities (SA)	**hours**
Learning using manual, course support, bibliography, course notes (SA)	
Additional documentation (in libraries, on electronic platforms, field documentation)	
Preparation for seminars/labs, homework, papers, portfolios and essays (mai mare sau egal cu nr. total ore prevăzut în calendarul disciplinei pentru temele de control)	
Tutorship	
Evaluations	
Other activities: [de ex.: comunicare bidirecţională cu titularul de disciplină / tutorele]	

3.7. Total individual study hours	
3.8. Total hours per semester	
3.9. Number of ECTS credits	**6**

4. Prerequisites (if necessary)

4.1. curriculum	*
4.2. competencies	*

5. Conditions (if necessary)

5.1. for the course	*
5.2. for the seminar /lab activities	*

6.1. Specific competencies acquired [1]

Professional/essential competencies	● create data models ● create software ● provide technical documentation
Transversal competencies	● show initiative ● assume responsibility ● think analytically

6.2. Learning outcomes

Knowledge	The student knows: traditional problem solving methods, heuristic approaches towards that cannot be solved properly using traditional methods, AI/ML/DL techniques for NP-complete problems
Skills	The student is able to code proper solutions in a programming language of choice and to optimize the problem solving activity towards finding the optimal solution to any given problem or at least an acceptable candidate solution for NP-complete problems
Responsibility and autonomy:	The student has the ability to work independently to obtain a good understanding of the input data of any problem, to devise a proper algorithm for solving the problem using the adoption, adaption and improvement of taught techniques to the given problem and to properly evaluate the results obtained after applying the algorithm to solve the given problem

7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	● introduce students to the realm of traditional, heuristic and AI based problem solving methods
7.2 Specific objective of the discipline	● teaching proper problem solving techniques catered to the current needs of the fields of computer science and mathematics

[1] One can choose either competences or learning outcomes, or both. If only one option is chosen, the row related to the other option will be deleted, and the kept one will be numbered 6.

8. Content

8.1 Course	Teaching methods	Remarks
Course 1. The Beginning 1. Information and Energy 2. Entropy 3. Continuum vs Discrete 4. Static vs Dynamic 5. Bits. Logics.Arithmetics.Algebra	Exposure: description, explanation, examples, discussion of case studies	
Course 2. Algebra and Geometry 1. Euclidean vs Noneuclidean Geometry 2. Design. Principles of Design 3. Numbers and Numerical Systems	Exposure: description, explanation, examples, discussion of case studies	
Course 3. Traditional Algorithms and Data Structures 1. Light. Wave-Particle Duality 2. Search 3. Data structures and their algorithms: a. linear b. alinear noncyclic c. cyclic 4. Algorithms and Programming 5. Hardware and Software: The Computer	Exposure: description, explanation, examples, discussion of case studies	
Course 4. Exhaustive vs Unexhaustive Search Methods 1. Should I Stay or Should I Go? 2. Traditional Algorithmic Methods: a. bruteforce/backtracking b. divide and conquer c. dynamic programming d. greedy 3. Heuristics	Exposure: description, explanation, examples, discussion of case studies	
Course 5. Outstanding Algorithms 1. Dijkstra. A* 2. Ford Fulkerson 3. Convolution 4. Fast Fourier Transform	Exposure: description, explanation, examples, discussion of case studies	
Course 5. Fundaments for Intelligence **1.** Decision Trees. Information Gain out of Entropy.Expert Systems **2.** NP Complete problems. Exploration vs Exploitation **3.** Hill Climbing. Tabu Search. Simulated Annealing **4.** Genetic Algorithms	Exposure: description, explanation, examples, discussion of case studies	
Course 6. Hive Intelligence 1. Communities: Ant Colony Optimization, Particle Swarm Optimization 2. Graphs and networks: The Traveling Salesman Problem	Exposure: description, explanation, examples, discussion of case studies	
Course 7. Making the machine learn 1. Central Limit Theorem. Statistics	Exposure: description, explanation, examples, discussion	

2. Monte Carlo Methods 3. Descending Gradient.Error.Loss	of case studies	
Course 8. Understanding Cognition 1. System theory 2. Modeling the environment 3. The neuron 4. Prey / Predator. Dynamic Systems	Exposure: description, explanation, examples, discussion of case studies	
Course 9. Machine Learning I 1. The Perceptron 2. Learning: a. data preprocessing and visualization b. training c. performance evaluation 3. Machine learning the basic logical and arithmetical operations	Exposure: description, explanation, examples, discussion of case studies	
Course 10.Machine Learning II 1. Neural networks 2. Backpropagation of errors 3. They do it with matrices	Exposure: description, explanation, examples, discussion of case studies	
Course 11. Deep Learning 1. Sight: Convolutional Neural Networks 2. Hearing: Recurrent Neural Network, Long Short Term Memory 3. Reinforcement Learning 4. Transformer Models	Exposure: description, explanation, examples, discussion of case studies	
Course 12. Homo Ludens 1. Game Theory 2. Agents 3. Winning Strategies	Exposure: description, explanation, examples, discussion of case studies	
Course 13.Quantum Logics 1. Fuzzy Logics 2. Dichotomy revisited: Quantum Bit 3. Quantum Logical Gates. The Quantum Computer	Exposure: description, explanation, examples, discussion of case studies	
Course 14. Philosophical conclusions and ethical perspectives	Exposure: description, explanation, examples, discussion of case studies	
Bibliography		
8.2 Seminar / laboratory	Teaching methods	Remarks
1. Modelling Entropy		
2. Particle Simulation		
3. Simulating Euclidean Geometry		
4. Implementing an ALU from logical gates		
5. Implementing searching algorithms		
6. Implementing Dijkstra and A* algotithms		
7. Implementing Fast Fourier Transform		
8. Implementing a Genetic Algorithm		
9. Implementing a Perceptron		
10. Implementing a Neural Network		
11. Implementing CNN layers		
12. Implementing an Agent		
13. Discussing Quantum Computers		
14. Implementing a complete Game		
Bibliography		

1. DONALD E. KNUTH, The Art of Computer Programming, Addison-Wesley, 1998
2. P. N. Klein, Coding the Matrix. Linear Algebra through Applications to Computer Science, Newtonian Press, 2013.
3. P.D. Lax, M.S. Terrell, Calculus with Applications, Springer, 2014.
4. S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, 1995
5. KERNINGHAN B.W., RITCHIE D.M The C Procramming Language -- Ansi C. Prentice Hall, 1988.

9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

- The graduate knows, understands and applies the basic concepts and the fundamental algorithms of Artificial Intelligence and is able to evaluate them based on metrics.
- The graduate knows and understands the concepts and the techniques of knowledge representation and is able to apply them for problem solving.
- The graduate knows and understands the mathematical foundations needed to develop intelligent algorithms and is capable of using them for algorithm implementation.
- The graduate is able to evaluate, both quantitatively and qualitatively, the performance of intelligent systems.
- The graduate is able to apply fundamental algorithms of Artificial Intelligence in order to solve real-world problems.
- The graduate knows and understands the mathematical foundations needed to develop intelligent algorithms and is capable of using them for algorithm implementation.
- The graduate is able to combine diverse information to formulate solutions and develop development ideas for new products and applications.

10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course		oral examination	60%
10.5 Seminar/laboratory		in lab/seminary activities extra homework	40%
10.6 Minimum standard of performance			

- Each student should at least be able to employ traditional problem solving techniques and heuristic methods for solving mathematics and informatics problems.
- A final grade of at least 5 cummulated from the points received both for lab/seminary activities and from the final oral examination from the exam session

11. Labels ODD (Sustainable Development Goals)[2]

[2] Keep only the labels that, according to the *Procedure for applying ODD labels in the academic process*, suit the discipline and delete the others, including the general one for *Sustainable Development* – if not applicable. If no label describes the discipline, delete them all and write „*Not applicable*.".

Not applicable.

Date: ...	Signature of course coordinator Lect.dr. Mircea Gabriel	Signature of seminar coordinator Lect.dr. Mircea Gabriel
Date of approval: ...		Signature of the head of department Assoc.prof.phd. Adrian STERCA