SYLLABUS

Public-Key Cryptography

University year 2025-2026

1. Information regarding the programme

1.1. Higher education institution	Babeş-Bolyai University
1.2. Faculty	Mathematics and Computer Science
1.3. Department	Mathematics
1.4. Field of study	Computer Science
1.5. Study cycle	Bachelor
1.6. Study programme/Qualification	Computer Science
1.7. Form of education	Full-time education

2. Information regarding the discipline

2.1. Name of the discipline	**Public-Key Cryptography**					Discipline code	**MLE0049**
2.2. Course coordinator					Prof. PhD. Septimiu Crivei		
2.3. Seminar coordinator					Prof. PhD. Septimiu Crivei		
2.4. Year of study	3	2.5. Semester	5	2.6. Type of evaluation	C	2.7. Discipline regime	DS

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

3.1. Hours per week	**3**	of which: 3.2 course	**2**	3.3 seminar/laboratory	**1**
3.4. Total hours in the curriculum	42	of which: 3.5 course	28	3.6 seminar/laborator	**14**
Time allotment for individual study (ID) and self-study activities (SA)					**hours**
Learning using manual, course support, bibliography, course notes (SA)					14
Additional documentation (in libraries, on electronic platforms, field documentation)					8
Preparation for seminars/labs, homework, papers, portfolios and essays					14
Tutorship					14
Evaluations					8
Other activities:					0
3.7. Total individual study hours	**58**				
3.8. Total hours per semester	**100**				
3.9. Number of ECTS credits	**4**				

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]

[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.

Professional/essential competencies	C1.5 Development of program units and corresponding documentation C3.3 Use of computer science and mathematical models and tools for solving specific problems in the application field
Transversal competencies	CT2 Efficient fulfillment of organized activities in an inter-disciplinary group and development of empathic abilities of inter-personal communication, relationship and collaboration with various groups

6.2. Learning outcomes

Knowledge	The student is able to ensure the formation of skills specific to the Mathematics and Algorithmics-related disciplines needed to complete the assignments. The student knows fundamental notions related to Cryptography, and methods of applying them to areas of science related to Mathematics and Computer Science.
Skills	The graduate will develop mathematical and algorithmical thinking, progressing from a procedural/computational understanding of mathematics to a broad understanding encompassing logical reasoning, generalization, abstraction, and formal proof.
Responsibility and autonomy:	The student is able explore some applied mathematical content independently, drawing on ideas and tools from previous coursework to extend their understanding. The student will independently extend applied mathematical ideas and arguments from previous coursework to a mathematical/computer science topic not previously studied.

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

7.1 General objective of the discipline	To present mathematical algorithms used in public-key cryptography.
7.2 Specific objective of the discipline	Number-theoretic and algebra algorithms will be studied and implemented in projects.

8. Content

8.1 Course	**Teaching methods**	**Remarks**
1. Classical cryptography. Examples	interactive exposure, explanation, didactical demonstration	

2. Algorithm complexity, elements of number theory	interactive exposure, explanation, didactical demonstration	
3. Public-key cryptography. RSA	interactive exposure, explanation, didactical demonstration	
4. Algorithms for testing primality	interactive exposure, explanation, didactical demonstration	
5. Algorithms for factoring integers	interactive exposure, explanation, didactical demonstration	
6. Quadratic residues. Rabin public-key cryptosystem	interactive exposure, explanation, didactical demonstration	
7. Polynomials. Finite fields	interactive exposure, explanation, didactical demonstration	
8. ElGamal public-key cryptosystem	interactive exposure, explanation, didactical demonstration	
9. Algorithms for computing discrete logarithms	interactive exposure, explanation, didactical demonstration	
10. Factorization of polynomials: Berlekamp's algortihm	interactive exposure, explanation, didactical demonstration	
11. Digital signatures	interactive exposure, explanation, didactical demonstration	
12. Key-related protocols	interactive exposure, explanation, didactical demonstration	
13. Practical aspects of public-key cryptosystems	interactive exposure, explanation, didactical demonstration	
14. Eliptic-curve cryptography	interactive exposure, explanation, didactical demonstration	

Bibliography

1. M. Cozzens, S.J. Miller, The Mathematics of Encryption: An Elementary Introduction, American Mathematical Society, 2013.
2. S. Crivei, A. Marcus, C. Sacarea, C. Szanto, Computational algebra with applications to coding theory and cryptography, Editura EFES, Cluj-Napoca, 2006.
3. C. Gherghe, D. Popescu, Criptografie. Coduri. Algoritmi, Editura Univ. Bucuresti, 2005.
4. A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, 1997. [http://www.cacr.math.uwaterloo.ca/hac]
5. C. Paar, J. Pelzl, Understanding Cryptography, Springer, 2009.

8.2 Seminar / laboratory	**Teaching methods**	**Remarks**
1. Classical cryptography	interactive exposure, algorithmization	The lab is scheduled as 2 hours every second week
2. Algorithm complexity	interactive exposure, algorithmization	
3. Modular arithmetics	interactive exposure, algorithmization	
4. Algorithms for testing primality	interactive exposure, algorithmization	
5. Algorithms for factoring integers	interactive exposure, algorithmization	
6. Public-key cryptography	interactive exposure, algorithmization	
7. Practical aspects of public-key cryptosystems	interactive exposure, algorithmization	

Bibliography

1. M. Cozzens, S.J. Miller, The Mathematics of Encryption: An Elementary Introduction, American Mathematical Society, 2013.
2. S. Crivei, A. Marcus, C. Sacarea, C. Szanto, Computational algebra with applications to coding theory and cryptography, Editura EFES, Cluj-Napoca, 2006.
3. C. Gherghe, D. Popescu, Criptografie. Coduri. Algoritmi, Editura Univ. Bucuresti, 2005.
4. A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, 1997. [http://www.cacr.math.uwaterloo.ca/hac]
5. C. Paar, J. Pelzl, Understanding Cryptography, Springer, 2009.

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 contents is directed towards practical applications of public-key cryptography. The topic is present in the computer science study programme of all major universities.

10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	Use of basic concepts in examples	Assessments	1/2 of the grade
10.5 Seminar/laboratory	Implement course concepts and algorithms	Practical examination	1/2 of the grade
10.6 Minimum standard of performance			
The final grade must be at least 5.			

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

	General label for Sustainable Development							
								

Date:
11.04.2025

Signature of course coordinator

Prof. PhD. Septimiu Crivei

Signature of seminar coordinator

Prof. PhD. Septimiu Crivei

Date of approval:
25.04.2025

Signature of the head of department

Prof. PhD. Andrei Mărcuş

[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*.”.