COURSE DESCRIPTION

Modular arithmetics and cryptography

Academic year 2026-2027

1. Programme-related data

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	Master
1.6. Study programme/Qualification	Data bases
1.7. Form of education	Full-time education

2. Course-related data

2.1. Course title	**Modular arithmetics and cryptography**			Course code	**MME3051**
2.2. Course coordinator	Prof. PhD. Septimiu Crivei				
2.3. Seminar coordinator	Prof. PhD. Septimiu Crivei				
2.4. Year of study 1	2.5. Semester 1		2.6. Type of assessment	Exam	
2.7. Course status	Optional		2.8. Course type	Core subject	

3. Total estimated time (hours per semester of teaching activities)

3.1. Number of hours per week	**3**	of which: 3.2. course	**2**	3.3. seminar/ laboratory/ project	**1**
3.4. Total of hours in the curriculum	42	of which: 3.5. course	28	3.6. seminar/ laboratory	**14**
Time allocation for individual study (IS) and self-taught activities (ST)					**hours**
Learning from textbooks, course materials, bibliography, and notes (IS)					28
Additional research in the library, on subject-specific electronic platforms, and on-site					28
Preparing seminars/ laboratories/ projects, assignments, reports, portfolios, and essays					28
Tutoring (professional guidance)					10
Examinations					14
Other activities					0
3.7. Total hours of individual study (IS) and self-taught activities (ST)				**108**	
3.8. Total hours per semester				**150**	
3.9. Number of credits				**6**	

4. Prerequisites (where applicable)

4.1. curriculum-related	
4.2 skills-related	

5. Specific conditions (where applicable)

5.1. course-related	
5.2. seminar/laboratory-related	

6.1. Competencies resulting from the completion of the degree programme *(as referred to in the curriculum)*[1]

Professional competencies	
Competency code	**Competency**
CP1	Advanced knowledge of theoretical, methodological, and practical developments in computer science
CP3	Use advanced skills to develop and conduct complex software projects, of practical and/or research nature, using a wide range of quantitative and qualitative methods
Transversal competencies	
Competency code	**Competency**
CT1	Systematic use of computer science knowledge to model and interpret new situations, within application contexts larger than the known ones
CT4	Team work abilities, assuming different execution and leading roles, performing professional tasks with considerable amounts of autonomy and responsibility

6.2. Learning outcomes relevant to the degree programme (as referred to in the curriculum)[2]

Learning outcomes targeted by the subject		
Competency code	**Knowledge and comprehension**	**Specific academic skills**
CF1	Absolventul/a posedă cunoştinţe fundamentale de modelare prin care analizează probleme din viaţa reală, le transpune în cerinţe concrete şi elaborează un model software corespunzător	Absolventul/a demonstrează abilităţi avansate de programare care vor permite acumularea de cunoştinţe solide şi înţelegerea rapidă a tehnologiilor moderne din domeniu

7. Subject-specific learning outcomes

Knowledge and comprehension
1. The student is able to ensure the formation of skills specific to the Mathematics and Algorithmics-related disciplines needed to complete the assignments.
2. The student knows fundamental notions related to Cryptography, and methods of applying them to areas of science related to Mathematics and Computer Science.
Specific academic skills
1. 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.

8. Contents

8.1. Course	**Teaching and learning methods**	**Remarks[3]**
1. Algorithm complexity, modular arithmetics	exposition, algorithmization	
2. Primality and factorization	exposition, algorithmization	

[1] The professional and/or transversal skills targeted by the subject for which the course description is prepared will be copied from the curriculum of the degree programme. For each competency, the complete entry, including the competency code, will be copied with the exact wording that appears in the curriculum, without any changes. If no competency is copied from either of the two categories, the row corresponding to that category is deleted from the table.

[2] The learning outcomes relevant for the degree programme and targeted by the subject for which the course description is prepared will be listed. The entries, copied without any changes from the Curriculum by subject type (Core Subject/Specialisation Subject/Complementary Subject), are listed under the corresponding competency.

[3] For example, organisational aspects, recommendations for students, specific aspects relating to the course/seminar, such as inviting experts in the field, etc.

3. Finite fields and discrete logarithms	exposition, algorithmization	
4. Classical cryptography	exposition, algorithmization	
5. DES, AES	exposition, algorithmization	
6. Stream ciphers	exposition, algorithmization	
7. Block ciphers	exposition, algorithmization	
8. RSA cryptosystem	exposition, algorithmization	
9. ElGamal cryptosystem	exposition, algorithmization	
10. Hash functions	exposition, algorithmization	
11. Digital signatures	exposition, algorithmization	
12. Key-related protocols	exposition, algorithmization	
13. Elliptic curve cryptography	exposition, algorithmization	
14. Quantum cryptography	exposition, 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.

8.2. Seminar/ laboratory	**Teaching and learning methods**	**Remarks**
1. Algorithm complexity, modular arithmetics	problematization, exercise	The seminar is scheduled as 2 hours every second week
2. Primality and factorization	problematization, exercise	
3. Finite fields and discrete logarithms	problematization, exercise	
4. Classical cryptography	problematization, exercise	
5. Block ciphers	problematization, exercise	
6. Public-key cryptography	problematization, exercise	
7. Digital signatures	problematization, exercise	

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. Evaluation

Type of activity	9.1 Evaluation criteria[4]	9.2 Evaluation methods[5]	9.3 Percentage in the final grade
9.4. Course	Use of basic concepts in examples	Presentation	1/3 of the grade
9.5. Seminar/ laboratory	Problem solving, project presentation	Test, practical examination	2/3 of the grade
9.6 Minimum standard for passing			
The final grade must be at least 5.			

10. SDG labels (Sustainable Development Goals)[6]

	○	Sustainable Development Generic Label

								
○	○	○	○	○	○	○	○	X
								No label applies
○	○	○	○	○	○	○	○	○

Date of entry:
15.04.2026

Signature of course coordinator

Prof. PhD. Septimiu Crivei

Signature of seminar coordinator

Prof. PhD. Septimiu Crivei

Date of approval in the department:
23.04.2026

Signature of the head of department

Prof. PhD. Andrei Mărcuş

[4] The evaluation criteria must directly reflect the learning outcomes targeted at the level of the degree programme respectively at the level of the subject. More specifically, the learning outcomes set out in the expected learning outcomes are assessed.

[5] Both final evaluation methods and ongoing evaluation strategies should be established.

[6] Select a single label which, according to the *Implementation of SDG labels in the academic process*, best matches the subject. If the subject addresses sustainable development in a generic manner (i.e. by presenting/introducing the general framework of sustainable development, etc.), then the Sustainable Development generic label may be applied. If none of the labels describe the subject, select the last option: *"No label applies."*