SYLLABUS

1. Information regarding the programme

1.1 Higher education institution	**Babeș-Bolyai University**
1.2 Faculty	**Faculty of Mathematics and Computer Science**
1.3 Department	**Department of Computer Science**
1.4 Field of study	**Mathematics**
1.5 Study cycle	**Bachelor**
1.6 Study programme / Qualification	**Mathematics Computer Science (in English)**

2. Information regarding the discipline

2.1 Name of the discipline (en) (ro)			**Intelligent methods and their applications in software engineering** **Metode inteligente și aplicațiile lor în ingineria software**				
2.2 Course coordinator			**Lect. PhD. Zsuzsanna Oneț-Marian**				
2.3 Seminar coordinator			**Lect. PhD. Zsuzsanna Oneț-Marian**				
2.4. Year of study	**3**	2.5 Semester	**6**	2.6. Type of evaluation	**E**	2.7 Type of discipline	**Optional**
2.8 Code of the discipline		MLE7040					

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

3.1 Hours per week	4	Of which: 3.2 course	2	3.3 seminar/laboratory	1 lab + 1 sem
3.4 Total hours in the curriculum	48	Of which: 3.5 course	24	3.6 seminar/laboratory	24
Time allotment:					Hours
Learning using manual, course support, bibliography, course notes					22
Additional documentation (in libraries, on electronic platforms, field documentation)					30
Preparation for seminars/labs, homework, papers, portfolios and essays					30
Tutorship					15
Evaluations					5
Other activities:					

3.7 Total individual study hours	102
3.8 Total hours per semester	150
3.9 Number of ECTS credits	6

4. Prerequisites (if necessary)

4.1. curriculum	• Programming Fundamentals, Object-Oriented Programming,

	Advanced Programming Methods
4.2. competencies	• Good programming skills in Python

5. Conditions (if necessary)

5.1. for the course	• Course hall with projector
5.2. for the seminar /lab activities	•

6. Specific competencies acquired

Professional competencies	CE1.3 Using intelligent methods, techniques and algorithms in order to model several classes of problems CE1.4 Identify and explain intelligent techniques and algorithms and using them to solve specific problems C4.3 Identifying appropriate models and methods to solve real problems
Transversal competencies	CT1. To apply the rules for organized and efficient work. CT2. To respect the principles and norms of professional etiquette. CT3. To use efficient learning methods and techniques for learning, documenting and searching CT4. To develop the capacity to use knowledge, adapt at the requests of a dynamic society, and properly communicate in English.

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

7.1 General objective of the discipline	• The goal of this course is to familiarize the students with intelligent search methods and clustering algorithms and how these can be used to solve different software engineering related problems.
7.2 Specific objective of the discipline	• Understand and recognize the components of a search/optimization problem. • Learn about different software testing types and understand how to describe software testing activities as search problems. • Recognize some of the most well-known bad smells in source code. • Understand clustering algorithms and how they can work with software engineering data.

8. Content

8.1 Course	Teaching methods	Remarks
1. Course organization. Search-based software engineering. Components of an optimization problem.	- Interactive exposure - Explanation. - Conversation - Didactical demonstration - Case studies	
2. Supervised and unsupervised learning. Software metrics.		
3. Hill climbing Simulated annealing		

Tabu search		
4. Genetic Algorithms		
5. Applications of search algorithms for unit testing		
6. Applications of search algorithms for fuzz testing		
7. Applications of search algorithms for regression testing I		
8. Applications of search algorithms for regression testing II		
9. Clustering algorithms.		
10. Clustering methods used for software testing		
11. Intelligent methods for software refactoring		
12. Software code embedding based approaches		

Bibliography

1. Mark Harman, Bryan F. Jones: Search-based software engineering, Information and software Technology, Nr. 43, pp. 833-839, 2001
2. Mark Harman, S. Afshin Mansouri, Yuanyuan Zhang, Search-based Software Engineering: Trends, Techniques and Applications, ACM Cmoputing Surveys, Vol. 45, Nr. 1, Article No. 11, pp. 1- 61, 2012
3. David Goldberg: genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Professional, 1989
4. Martin Fowler: Refactoring. Improving the design of Existing Code, Addison-Wesley Professional, 2018

8.2 Seminar	Teaching methods	Remarks
1. Seminar organization	- Conversation - Dialog - Case studies	- Seminar will be organized as 2 hours every two weeks. - During the seminars, each student will present a theoretical report.
2. Theoretical report topic selection		
3. Theoretical report paper selection		
4. Theoretical report presentation + discussions		
5. Theoretical report presentation + discussions		
6. Theoretical report presentation + discussions		

Bibliography

1. Mark Harman, Bryan F. Jones: Search-based software engineering, Information and software Technology, Nr. 43, pp. 833-839, 2001
2. Mark Harman, S. Afshin Mansouri, Yuanyuan Zhang, Search-based Software Engineering: Trends, Techniques and Applications, ACM Cmoputing Surveys, Vol. 45, Nr. 1, Article No. 11, pp. 1- 61, 2012
3. David Goldberg: genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Professional, 1989
4. Martin Fowler: Refactoring. Improving the design of Existing Code, Addison-Wesley Professional, 2018

8.3 Laboratory	Teaching methods	Remarks
1. Python libraries that will be used during the labs	- Presentation - Conversation - Dialog - Case studies - Team work - Exercises	- Lab will be organized as 2 hours every two weeks. - During every lab, students will work in 2-3 person teams and will have to solve simple problems in Python related to the
2. Search algorithms in Python		
3. Search algorithms for unit / fuzz testing		
4. Clustering algorithms in Python		
5. Software refactoring		
6. Intelligent methods for software restructuring		

		topics discussed at the lecture.

Bibliography

1. Mark Harman, Bryan F. Jones: Search-based software engineering, Information and software Technology, Nr. 43, pp. 833-839, 2001
2. Mark Harman, S. Afshin Mansouri, Yuanyuan Zhang, Search-based Software Engineering: Trends, Techniques and Applications, ACM Cmoputing Surveys, Vol. 45, Nr. 1, Article No. 11, pp. 1- 61, 2012
3. David Goldberg: genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Professional, 1989
4. Martin Fowler: Refactoring. Improving the design of Existing Code, Addison-Wesley Professional, 2018

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 content of the discipline is consistent with the similar disciplines from other Romanian universities and universities from abroad, as well as with the requirements that potential employers would have in the software engineering field.

10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	Knowledge of basic notions about search and clustering algorithms and their applications in software engineering.	Written exam in the exam session	40%
10.5. Seminar	Capability of recognizing and understanding the discussed topics in a research paper.	Presentation of the content of one research paper	20%
10.6 Lab activities	Correctness and completeness of the lab projects solved in 2-3 person teams.	Continuous observation during the labs. Average grade of the 6 lab projects.	40%
10.6 Minimum performance standards			

- Each students needs to demonstrate that he/she acquired an acceptable level of knowledge and understanding of the domain and that he/she is capable of coherently expressing this knowledge.
- Written exam grade should be greater than 5 and final grade should be greater than 5.
- At least 5 attendances are mandatory at the labs and 4 at the seminars, otherwise the student is not allowed to participate at the written exam.
- At least one (out of six) lab project should be solved in order to be able to participate at the written exam.

Date	Signature of course coordinator	Signature of seminar coordinator
29.04.2024	Lect. PhD. Zsuzsanna Oneț-Marian	Lect. PhD Zsuzsanna Oneț-Marian

Date of approval ..... Signature of the head of department

Conf. PhD. Adrian Sterca