SYLLABUS

1. Information regarding the programme

1.1 Higher education institution	**Babeş Bolyai University**
1.2 Faculty	**Faculty of Mathematics and Computer Science**
1.3 Department	**Department of Computer Science**
1.4 Field of study	**Computer Science**
1.5 Study cycle	**Master**
1.6 Study programme / Qualification	**Cyber Security**

2. Information regarding the discipline

2.1 Name of the discipline (en) (ro)	**Program Analysis for Software Security** **Analiza Programelor pentru Securitatea Software**						
2.2 Course coordinator	**Assoc. Prof. Eng. Florin Craciun**						
2.3 Seminar coordinator	**Assoc. Prof. Eng. Florin Craciun**						
2.4. Year of study	**2**	2.5 Semester	**3**	2.6. Type of evaluation	**E**	2.7 Type of discipline	**Mandatory**
2.8. Code of the discipline	MME8201						

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	1sem + 1pr
3.4 Total hours in the curriculum	56	Of which: 3.5 course	28	3.6 seminar/laboratory	28
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					20
Additional documentation (in libraries, on electronic platforms, field documentation)					24
Preparation for seminars/labs, homework, papers, portfolios and essays					35
Tutorship					5
Evaluations					10
Other activities:					-

3.7 Total individual study hours	94
3.8 Total hours per semester	150
3.9 Number of ECTS credits	6

4. Prerequisites (if necessary)

4.1. curriculum	• None
4.2. competencies	• Basic software development skills

5. Conditions (if necessary)

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

6. Specific competencies acquired

Professional competencies	• Understanding and working with basic concepts in software engineering; • Capability of analysis and synthesis; • Proficient use of methodologies and tools specific tool software systems • Organization of software production processes.
Transversal competencies	• Team work capabilities; able to fulfill different roles • Professional communication skills; concise and precise description, both oral and written, of professional results, • Antepreneurial skills;

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

7.1 General objective of the discipline	• be able to apply basic methods for software formalization
7.2 Specific objective of the discipline	• Be able to write formal specifications • understanding of program verification • be able to use software verification tools

8. Content

8.1 Course	Teaching methods	Remarks
1. Introduction in program verification: main concepts	Exposure,description, explanation, debate and dialogue, discussion of case studies	
2. Formal Specifications	explanation, debate and dialogue, discussion of case studies	
3. Semantic models: Operational Semantics, Denotational Semantics	Exposure,description, explanation	
4. Logic: basic concepts, inference rules	Exposure,description, explanation	
5. Hoare logic: basics, weakest precondition	Exposure,description, explanation, discussion of case studies	
6. Hoare Logic: loops, invariants	Exposure,description, explanation, discussion of case studies	
7. Hoare Logic: modular verification	Exposure,description, explanation,	
8. Separation logic: introduction	Exposure,description, explanation	

9. Separation logic: inductive predicates, lemmas	Exposure,description, explanation, discussion of case studies	
10. Separation logic: entailment	Exposure,description, explanation, discussion of case studies	
11. Separation logic for object-oriented paradigm	Exposure,description, explanation, discussion of case studies	
12. Separation logic: arrays	Exposure,description, explanation, discussion of case studies	
13. Concurrent Separation Logic	Exposure,description, explanation, discussion of case studies	
14. Concurrent Separation Logic	Exposure,description, explanation, discussion of case studies	
Bibliography 1. Hoare logic research papers 2. Separation logic research papers		
8.2 Seminar / laboratory	Teaching methods	Remarks
1. Research papers allocation for the oral presentation	Use practical tools to implement group projects. Discuss research papers.	Seminar is organized as a total of 14 hours – 2 hours every second week Project is organized as a total of 14 hours – 2 hours every
2. Hoare Logic project allocation	Use practical tools to implement group projects. Discuss research papers.	
3. Research papers presentations	Use practical tools to implement group projects. Discuss research papers.	
4. Separation Logic project allocation	Use practical tools to implement group projects. Discuss research papers.	
5. Research papers presentations.	Use practical tools to implement group projects. Discuss research papers.	

6. Hoare Logic project presentation	Use practical tools to implement group projects. Discuss research papers.	
7. Separation Logic project presentation	Use practical tools to implement group projects. Discuss research papers.	
Bibliography verification tools Research papers		

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 course respects the IEEE and ACM Curriculla Recommendations for Software Engineering studies;
- The content of the course is considered by the software companies as important for average software development skills

10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	- know the basic principle of the domain; - apply the course concepts - problem solving	Written exam	50.00%
10.5 Seminar/lab activities	- be able to implement course concepts - e – be able to use verification tools - - be able to do a critical evaluation of research papers - to be able to write a critical essay	-Practical examination	50.00%
10.6 Minimum performance standards			
➢ At least grade 5 (from a scale of 1 to 10) at both written exam and laboratory work.			

Date Signature of course coordinator Signature of seminar coordinator

..... Assoc. Prof. En. Florin CRACIUN Assoc. Prof. Eng. Florin CRACIUN

Date of approval Signature of the head of department

..... Assoc. Prof. PhD. Adrian Sterca