Study programme(s): Informatics (IM)

Level: master

Course title: Software Testing (IB321)

Lecturers: Zoran D. Budimac, Danijela D. Tešendić

Status: elective

ECTS: 7,5

Requirements: none

Learning objectives

The course aims to present and critically analyze the current software testing techniques, particularly the importance of formal methods in testing.

Learning outcomes

Minimum: Students should be able to critically assess the importance of software testing and the need and usefulness of formal methods during testing.

Optimal: Students should be able to develop an integrated approach to software testing and formal theories.

Syllabus

Theoretical instruction

Theoretical basis for testing, structural testing, functional testing, the basis for combining formal methods and testing, formal methods based on the model, testing using finite state machines, testing using process algebra, testing using algebraic specification, mutation testing, testing using UML dynamic models, temporal logic and model of checking models and their role in testing and the process of managing software testing.

Practical instruction

Analysis of study examples.

Literature

1. C. Kaner, J. Falk, H. Q. Nguyen: Testing Computer Software, Wiley, 1999
2. B. Beizer, Software Testing Techniques, International Thomson Press, 1990
3. P. C. Jorgensen, Software Testing: A Craftsman's Approach, second edition, CRC Press, 2004
4. Edmund M. Clarke, Jr., Orna Grumberg and Doron A. Peled, Model Checking, MIT Press, 1999.
5. Ilene Burnstein. Practical Software Testing. Springer-Verlag, 2003
6. Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press, 2008.

Weekly teaching load				Other:
Lectures: 2	Exercises: 3	Other forms of teaching:	Student research:	

Teaching methodology

Classical methodology is applied in lectures with the use of a beam-projector. Exercises are conducted in traditional teaching methods, with the use of a beam-projector, and serve to analyze study examples and practical exercises on computers to practice the principles introduced and get familiar with the recommended tools. Students build their knowledge by researching the topics, whereby knowledge is checked through producing the papers that are presented during and at the end of the course.

Grading (maximum number of points 100)

Pre-exam obligations	**Points**	**Final exam**	**points**
Tests	**40**	Oral exam	**40**
Seminar(s)	**20**		