

Study programme(s): Applied Mathematics (MAP)		
Course title: PROGRAMMING 2 (P501)		
Lecturer(s): Sanja Rapajić		
Course status: compulsory		
ECTS points: 6		
Requirements: Programming 1		
Learning Objectives Acquisition of knowledge and skills of advanced programming, adoption of principles and techniques of object-oriented programming, training in utilization of modern packages in application development. Mastering the specifics and means of using abstract data structures, including dynamic structures, working with files, and capabilities of a graphical interface design.		
Learning Outcomes <i>Minimal:</i> At the end of the course, a successful student is expected to comprehend the basic concepts of advanced and object-oriented programming, and to demonstrate the ability to understand problems and realize their solutions in a specific programming language. <i>Desirable:</i> At the end of the course, a successful student is expected to identify suitable data structures for problem solving with a deep understanding and analysis of less standard problems and realization of their solutions.		
Syllabus <i>Theoretical instructions</i> Program structure, functions. Object-oriented programming, classes, inheritance, fields, methods. Abstract data structures — stacks, queues, lists, trees. References, reference types. Dynamic programming. Introduction to packages and libraries, applications to solving mathematical related problems. Working with files - reading, writing. Graphical user interfaces. <i>Practical instructions</i> Training and mastering the notions adopted in theoretical instructions through solving specific problems and their implementation on a computer.		
Literature 1. Gutttag, John. Introduction to computation and programming using Python: With application to understanding data . MIT Press, 2016. 2. Michael Dawson. Python: uvod u programiranje , prevod 3. izdanja, Mikro knjiga, 2010.		
Number of active classes	Lectures: 2	Exercises: 3
Teaching methods Classical teaching methods are used in lectures. Practical teaching in a computer lab, where students independently apply mastered techniques by developing various applications with an increasing complexity and capability as the semester passes. Knowledge is tested through colloquia and practical tasks.		

Grading (maximum number of points 100)			
Pre-exam obligations	Points	Final exam	Points
practical tasks	20	oral exam	40
colloquia	40		