

Study programme(s): Mathematics (M3)				
Level: (bachelor)				
Course title: Programming 1 M3-04				
Lecturer: Miloš Z. Stojaković				
Status: obligatory				
ECTS: 8				
Requirements: none				
Learning objectives Students should be capable of solving mathematical and programming problems. They should master algorithmic thinking and be able to define problems precisely, using object-oriented programming, integrated development environment (IDE) and debugger.				
Learning outcomes <i>Minimal:</i> Developing habits of algorithmic thinking in solving and interpretation of problems. Learning and using the basic data types and data structures, as well as basic algorithms suitable for solving mathematical problems. <i>Desirable:</i> Understanding concepts of procedural and object-oriented programming. Using arrays, collections, data structures and classes. Applying object-oriented programming techniques to solve mathematical problems. Knowing how to tackle combinatorial problems using advanced programming techniques.				
Syllabus <i>Theoretical instruction</i> Identifiers, commands. Variables, primitive data types, expressions. Basic control and repetitive structures. Using classes and objects. Stack and heap. Arrays and collections. Sorting. Extreme elements. Iterative and recursive approach. Dealing with basic combinatorial structures. <i>Practical instruction</i> Understanding and mastering the basic principles of programming. Using control and repetitive structures, as well as basic programming techniques. Implementation of algorithms for solving concrete mathematical problems, and modification of standard algorithms to solve new problems.				
Literature 1. John Sharp, Microsoft Visual C# 2005 korak po korak, CET (Microsoft Press), Beograd, 2006. 2. Miloš Stojaković, Skripte za predmet Programiranje 1, Novi Sad, 2012. 3. Microsoft Developer Network online documentation, msdn.microsoft.com				
Weekly teaching load				Other:
Lectures: 3	Exercises: 3	Other forms of teaching:	Student research:	
Teaching methodology Lectures are conducted in standard lecturing approach, sometimes supported by a projector. Exercises are aimed at practising the techniques acquired in lectures and discussing the possible applications of the programming methods on concrete problems, possibly including a modification of the algorithm used. Knowledge is tested in two colloquia, while in practical part of lectures students solve tests and are able to obtain some points.				
Grading (maximum number of points 100)				
Pre-exam obligations	points	Final exam	points	
Active participation in lectures		Written exam		
Practical instruction	10	Oral exam	40	
Colloquia	50			
Seminar(s)				