

**Faculty of Science and Mathematics / COMPUTING AND INFORMATION TECHNOLOGY /
INTRODUCTION TO COMPUTER SCIENCE**

Course:	INTRODUCTION TO COMPUTER SCIENCE			
Course ID	Course status	Semester	ECTS credits	Lessons (Lessons+Exercises+Laboratory)
1339	Mandatory	1	7	3+3+0
Programs	COMPUTING AND INFORMATION TECHNOLOGY			
Prerequisites	None			
Aims	Introducing basic algorithms and programming concepts using Java programming language: variables, types, loops, conditional statements, methods, classes. Learn how computer represents integer and real numbers, characters, pictures, audio, and video.			
Learning outcomes	At the end of the course, the participant is expected to be able to: 1. Analyze problems and develop computer algorithms to solve novel problems [Familiarity] 2. Describe the internal representation of numeric and non-numeric data [Familiarity] 3. Write, document, test and debug Java programs, making use of variables, expressions, selection and looping statements. [Usage] 4. Identify and correct syntax and logic errors in programs [Familiarity] 5. Organize program code into modules using methods following the software engineering principles of modularity and abstraction. [Usage] 6. Assemble data and methods into classes at an introductory level following the software engineering principles of encapsulation and data hiding. [Usage] 7. Effectively use Internet services and word processing software, spreadsheet software, and presentation software [Usage]			
Lecturer / Teaching assistant	Goran Šuković, Dušica Slović			
Methodology	This is a seven-credit course with six 45-minute sessions per week (three lecture sessions and three lab session). All lecture hours are face-to-face and involve a mixture of traditional lectures, interactive programming sessions, and “hands-on” learning activities			
Plan and program of work				
Preparing week	Preparation and registration of the semester			
I week lectures	Introduction to course. Integrated development environment.			
I week exercises	Intro to labs. Number systems. Integer representation.			
II week lectures	Java programs. Variables, types, constants, assignment.			
II week exercises	Data representation: IEEE 754 floating point.			
III week lectures	Arithmetic and logical expressions.			
III week exercises	Data representation: pictures, audio, video.			
IV week lectures	Conditional statements.			
IV week exercises	Operating system. Folders and files. Copying and moving files. Linking documents. External devices. Multimedia.			
V week lectures	Loops.			
V week exercises	Test. Computer security. Antivirus and antispyware. Firewall. Printer settings. Network settings. Sharing printers and files. Command prompt.			
VI week lectures	Methods.			
VI week exercises	Intro to text processing. Creating and formatting text documents.			
VII week lectures	Midterm.			
VII week exercises	Advanced text formatting. Tables, formulas, drawings, special characters.			
VIII week lectures	Simple algorithms.			
VIII week exercises	Templates, mail merge. Linking text documents.			
IX week lectures	Number theory algorithms.			
IX week exercises	Test.			
X week lectures	Arrays.			
X week exercises	Spreadsheets – creating and basic formatting.			
XI week lectures	Arrays – sorting and searching.			

ECTS catalog with learning outcomes
University of Montenegro

XI week exercises	Spreadsheets – advanced formulas and graphs. Pivot tables. Goal seek. Macros..					
XII week lectures	Two-dimensional arrays					
XII week exercises	Presentation software. Creating and formatting presentation.					
XIII week lectures	Introduction to object-oriented programming. Classes, fields, methods.					
XIII week exercises	Adding animation and sound effects. HTML basics.					
XIV week lectures	Object-oriented programming. Data encapsulation.					
XIV week exercises	Test.					
XV week lectures						
XV week exercises						
Student workload	5x40/30 = 6 sati 20 min Weekly: Lectures: 2 hours 15 min, Labs: 2 hours 15 min Other: 0, Individual work: 1 hours 50 min.					
Per week			Per semester			
7 credits x 40/30=9 hours and 20 minuts 3 sat(a) theoretical classes 0 sat(a) practical classes 3 excercises 3 hour(s) i 20 minuts of independent work, including consultations			Classes and final exam: 9 hour(s) i 20 minuts x 16 =149 hour(s) i 20 minuts Necessary preparation before the beginning of the semester (administration, registration, certification): 9 hour(s) i 20 minuts x 2 =18 hour(s) i 40 minuts Total workload for the subject: 7 x 30=210 hour(s) Additional work for exam preparation in the preparing exam period, including taking the remedial exam from 0 to 30 hours (remaining time from the first two items to the total load for the item) 42 hour(s) i 0 minuts Workload structure: 149 hour(s) i 20 minuts (courses), 18 hour(s) i 40 minuts (preparation), 42 hour(s) i 0 minuts (additional work)			
Student obligations						
Consultations			Room 128,			
Literature			J.G. Brookshear - Computer Science: An Overview, Addison Wesley, 2011. M. Martinović, P. Stanišić - Principi programiranja, Univerzitet Crne Gore, 2004. Herbert Schildt - Java JDK 7: kompletan priručnik (Mikro knjiga, 2012) Lecture slides. (PDF, PPT)			
Examination methods			3 test, 5 points each (15 points) - Essay and presentation. 5 points each (10 points) - midterm 35 points - Final 40 points.			
Special remarks			The lecturer is able to offer course in English and Russian.			
Comment			www.pmf.ac.me, uvod@rc.pmf.ac.me			
Grade:	F	E	D	C	B	A
Number of points	less than 50 points	greater than or equal to 50 points and less than 60 points	greater than or equal to 60 points and less than 70 points	greater than or equal to 70 points and less than 80 points	greater than or equal to 80 points and less than 90 points	greater than or equal to 90 points