NAME OF THE COURSE	Programming in the profession I					
Code	PMP073	Year of study	2 PD			
Course teacher	Toni Šćulac	Credits (ECTS)	4,0			
Associate teachers		Type of instruction (number of hours)	P	S	V	T
			30	0	30	
Status of the course	Obligatory	Percentage of application of e-learning	10%			

COURSE DESCRIPTION	
Course objectives	This course provides an introduction to the C programming language with applications to mathematical and physical problems.
Course enrolment requirements and entry competences required for the course	Basic knowledge in physics and mathematics.
Learning outcomes expected at the level of the course (4 to 10 learning outcomes)	On completion of this course a student should be able to: 1. Understand the basic terminology used in computer programming applied to C programming language. 2. Design, write, compile and run programs written in C. 3. Use different data types, understand the structure and control flow of program. 4. Implement and use function in some simple cases.
Course content broken down in detail by weekly class schedule (syllabus)	1. Introduction to C programming. Structure of a program, compiling and execution. Standard input and output. 2. Structure of a program. Directives, functions, commands, macro. Keywords, tokens. 3. Constants, variables, data types. 4. Communication with a program. printf and scanf. 5. Operators and expression. Evaluation of expressions, operator precedence and associativity. 6. Decision making and branching: if, else, switch, break 7. Loops: for, while, do, break, continue, goto 8. math.h library. Series implementation with loops. 9. Midterm 10. Data types in C. Numerical accuracy and data types. Data type conversion. Memory, overflow, reservators, escape sequence, typdef, sizeof. 11. Arrays. Multidimensional arrays. Variable-length arrays. Constant arrays. Array initialization and declaration. 12. Functions: definition, call, declaration. Arguments and parameters. Value passing, return and exit. 13. Pointers: memory organization, address, declaration, assignment, operators. 14. File access. 15. Elective topics: pointers, C libraries, intro to numerical analysis
Format of instruction	lectures assignments exercises

Student responsibilities	Active participation in classes and assignments. Solving given problems in class. Homework.
Screening student work *(name the proportion of ECTS credits for each activity so that the total number of ECTS credits is equal to the ECTS value of the course)*	Lectures attendance 1.5 ECTS Colloquium 1.0 ECTS Practical work 1.5 ECTS
Grading and evaluating student work in class and at the final exam	Evaluation is based on activity, written exams and homework assignments.
Required literature (available in the library and via other media)	1. Lecture notes in Programming in profession, Hrvoje Kalinić 2. B. W. Kernighan & D. M. Ritchie "The C programing langauge" , Prentice Hall, USA, 1998. 3. Mike Banahan, Declan Brady, Mark Doran "The C book", GBDirect, 2008.
Optional literature (at the time of submission of study programme proposal)	1. Robert Sedgewick, " Algorithms in C", Addison-Wesley, USA, 1998. 2. K. Ćosić, P. Marendić, "Naučite programirati uz C++", Element, 2009.
Quality assurance methods that ensure the acquisition of exit competences	Students feedback, students results and self-evaluation.
Other (as the proposer wishes to add)	