Course unit title	Course unit code
INTRODUCTION TO PROGRAMMING	

Lecturer (s)	Department where course unit is delivered
Assoc Prof. Dr. Vytautas Rudžionis	Kaunas Faculty Institute of Social Sciences and Applied Informatics

Cycle	Level of course unit	Type of the course unit
First	1/1	Compulsory

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	1 semester 09-01 – 01-26	Lithuanian

Prerequisites and corequisites	
Prerequisites: No	**Corequisites:**

Number of ECTS credits allocated	Student‘s workload	Contact work hours	Individual work hours
5	130	52	78

Purpose of the course unit: programme competences to be developed		
To acquire the ability properly apply main elements of programming languages, program structures, standard libraries; to be able to develop simple software applications, to able to identify strong and weak points of software code from security point of view.		
Learning outcomes of course unit	**Teaching and learning methods**	**Assessment methods**
Will be able transform the given algorithm to software code, will be able to select best tools to implement the given algorithm	Formal lecture, Practical exercise Individual assignments Active teachning methods (programming, algorithm analyzis)	Control assignment; independent software ogramming and defending the applied methods

Course content: breakdown of the topics	Contact work hours							Individual work hours and tasks	
	Lectures	Consultations	Seminars	Practice classes	Laboratory	Practice	**All contact work**	**Individual work**	**Tasks**
Algorithm and programm. Relation between algorithm and programm	2				4		**6**	**10**	Software coding

Main elements of programming languages: variables, data types, expressions, operations, program control	2				4		**6**	**10**	Software coding
Main elements of programming languages (2): branching operators, condition operators, loops, switch operator	2				8		**10**	**10**	Software coding, preparation for control assignment
Arrays, strings, memory control	2				4		**6**	**10**	Software coding
Functions: sunroutines and functions, definition of functions, types of functions, structural programming	2				8		**10**	**10**	Software coding,
Standard libraries and functions: library, standard function, file processing, input.output operations, control of computer devices	6				4		**10**	**20**	Software coding, preparation for control assignment
Consultation		2					**2**	**8**	
Exam						2	**2**		
Total	**16**	2			**32**	**2**	**52**	**78**	

Assesment strategy	**Comparative weight percentage**	**Date of examination**	**Assesment criteria**
I control assignment	15%	At predefined time	Student gets task and needs to write code to realize the task in one hour. Criteria taken into consideration: - accuracy of algorithm; - accuracy of code; - efficiency of code
II control assignment	15 %	At predefined time	Student gets task and needs to write code to realize the task in one hour. Criteria taken into consideration: - accuracy of algorithm; - accuracy of code; - efficiency of code
Individual assignment, defending the proposed solution ID	20%	At predefined time	Student receives freely formulated task and needs to develop algorithm for solution and to write program in selected programming language Graded in 1-10 mark scale. **10-9:** Perfect and very good

			knowledge. Evaluation level. 90-100 % correct answers.. **8-7:** Good knowledge and abilities could be several mistakes. Synthesis level. 70-89 % correct answers. **6-5:** Average knowledge and abilities, there are errors. Analysis level. 50-69 % correct answers. **4-3:** Knowledge and abilities below average, there are significant errors. Knowledge application level. 20-49 % correct answers. **2-1:** Below minimum requirements. 0-19 % correct answers.
Exam -E	50 %	Assigned time during exam session	Test contains 10 questions of different complexity (varies from understanding of algorithm to knowledge of programming techniques). Graded in 1-10 mark scale. **10-9:** Perfect and very good knowledge. Evaluation level. 90-100 % correct answers.. **8-7:** Good knowledge and abilities, could be several mistakes. Synthesis level. 70-89 % correct answers. **6-5:** Average knowledge and abilities, there are errors. Analysis level. 50-69 % correct answers. **4-3:** Knowledge and abilities below average, there are significant errors. Knowledge application level. 20-49 % correct answers. **2-1:** Below minimum requirements. 0-19 % correct answers.
Exam (E) include all materials (grade E = E if E >= 5, else E=0). Final grade is calculated as follows: Balas =Exam*0,5+I control assignment*0,15+ II contril asignment*0.15+Individual assignment*0.2.			

Author	Year	Title	Number of periodical publication or publication Volume	The place of publication and publisher or online link
Required reading				
Halterman R.	2015	Fundamentals of C++ Programm		https://tfetimes.com/wp-content/uploads/2015/04/progcpp.pdf

		ing		
Morin P.	2011	Open Data Structures		https://tfetimes.com/wp-content/uploads/2015/04/ods-cpp.pdf
Backman K.	2012	Structured Programming in C++		https://tfetimes.com/wp-content/uploads/2015/04/structured-programming-with-c-plus-plus.pdf
Recommended reading				
Felleisen M., Findley R., Flatt M., Krishnamurti S..	2003	How to Design Programs		Boston, MIT Press
Bentley J.	2000	Programming Pearls		New York, Addison-Wesley