

Course unit title	Course unit code
PROGRAMMING LANGUAGES AND OBJECT ORIENTED 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/Individual Studies

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	Spring semester 02-01 – 06-26	Lithuanian

Prerequisites and corequisites	
Prerequisites: Algorithms, data structures, computer architectures, introduction to programming	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 be able to code in high level languages, to be able analyze and understand programming paradigms (procedural and object oriented), to acquire the ability understand, analyze and apply object oriented programming principles, to be able develop not complicated software applications.		
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 teaching methods (programming, algorithm analysis)	Control assignment; independent software programming and defending the applied methods
Will be able to apply principles of object oriented programming to given algorithm, will be able to define most efficient tools for algorithm realization	Formal lecture, Practical exercise Individual assignments Active teaching methods (programming, algorithm analysis)	Exam; independent software programming and defending the applied methods

Course content: breakdown of the topics	Contact work hours	Individual work hours and tasks
---	--------------------	---------------------------------

	Lectures	Consultations	Seminars	Exam	Laboratory	Practice	All contact work	Individual work	Tasks
Programming languages. Classification of programming languages. Low and high level languages. Main components of programming languages.	2				4		6	10	Software coding
Most popular programming languages: C++, Java, C#. Specific features of those languages.	2				4		6	10	Software coding
Fundamentals of object oriented programming: object; class; object oriented programming; encapsulation, inheritance; advantages of object oriented programming	4				8		12	10	Software coding, preparation for control assignment
Fundamentals of object oriented programming: polymorphism, templates, virtual functions, examples of objects	4				8		12	20	Software coding
Patterns	2				4		6	10	Software coding, preparation for control assignment
Principles of automatic code generation: advantages of code's reusability, automatic code generation; limits of automatic code generation; tools for automatic code generation	2				4		6	18	Software coding
Consultation		2					2		
Exam				2			2		
Total	16	2		2	32		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				
Horstmann C.	2010	OO Design and Patterns		New York: Wiley
Budd T.	2012	An Introduction to Object Oriented Programmin g		O'Reilly Media
Bugayenko E.	2015	Elegant Objects		CreateSpace Independent Publishing Platform
Recommended reading				
Horstmann C., Budd T.	2008	Big C++		New York: Wiley
Liberty J., McDonald B.	2008	Learning C#		Boston, O'Riley

