

## OBJECT ORIENTED PROGRAMMING IN C++

Title of Study Programme and Code		Type (compulsory/optional)	Cycle	Year of study when the component is delivered (if applicable)
Information Systems Engineering 6531EX043		Compulsory	1 <sup>st</sup>	1 <sup>st</sup> year
Semester/trimester when the component is delivered		Number of ECTS credits allocated	Language of instruction	Mode of delivery (face-to-face/e-learning/...)
2 <sup>nd</sup>		5 ECTS	English	Face-to-face/e-learning
Learning outcomes			Study methods	Assessment methods
After completion of the study subject, a student should be able to:			Interpretation of new concepts (terms); Review of visual material; Interactive lecture; Demonstration; Discussion; Practical works; Group work; Literature studies; Analysis of solved tasks.	Assessment of control work; Assessment of practical work; Assessment of practical group work; Assessment of presentation and defense of course work; Assessment of exam tasks.
<b>LO 1</b>	Know the principles of object-oriented programming and to apply it.			
<b>LO 2</b>	Know features of C ++ syntax.			
<b>LO 3</b>	Install an IDE for C ++ programming language.			
<b>LO 4</b>	Develop programs in C ++ programming language.			
<b>LO 5</b>	Ensure the stability and reliability of designed program.			
<b>LO 6</b>	Install developed software.			
Prerequisites (these courses must be successfully completed prior to taking this particular course)				
-				
Course content				
1. Programming paradigms and their differences. Object-oriented programming. Installing of the IDE for C++. 2. Type of structure data. Recursion. Pointers. 3. Files. Input / Output Flows. Text analysis and handling. 4. Object-oriented programming concepts: Classes and objects. Encapsulation. Friendliness and inheritance. Polymorphism. Constructor and destructor. Virtual methods. Templates. 5. Abstract data type. Creating a common class structure. 6. Classes with dynamic data fields. 7. Program Testing. Mistakes. Exception. Exception processing. Security. Preparing the installation package. 8. Standard Template Library (STL). 9. User Interface. Graphical User Interface Modeling.				
Recommended or required reading and other learning resources/tools				
1. J. Urbonienė. Course in VLE Moodle. Available at: <a href="https://moodle.utenos-kolegija.lt/">https://moodle.utenos-kolegija.lt/</a>				

2. Code::Blocks: <http://www.cplusplus.com/doc/tutorial/introduction/codeblocks/>
3. C++ Language: <http://www.cplusplus.com/doc/tutorial/>
4. Code::Blocks student manual:  
<http://www.sci.brooklyn.cuny.edu/~goetz/codeblocks/codeblocks-instructions.pdf>