I048	**Object-Oriented Programming**	L	S	E	ECTS
		2	0	2	6

Course objectives. Course objectives are to familiarize students with basic concepts of object oriented programming in C++ programming language, and with SOLID concepts of object oriented design. Students will acquire knowledge that will help them work in a team, as well as to individually approach problems. Moreover, students will learn basic concepts of parallel programming in multiple processes and threads (C++ threads library).

Course prerequisites. Not required.

Syllabus.

1. Data encapsulation: Class. Public, private and protected class members. Class methods. Constructors and destructors. Overloading of constructor and operators. Pointer to classes. Key word "this". Static class members. Dynamic allocation of objects.
2. Friendship and inheritance: Friend functions. Friend classes. Inheritance. Multiple inheritance.
3. Polymorphism: Pointer to the base class. Virtual class members. Abstract base class.
4. SOLID principles in object oriented design (examples). Lambda function vs. functor. Decltype. Static assert. Variadic templates.
5. Linux. Compiling with gcc compiler. Makefile. Modern C++ standards (c++11, c++14). Git version control system.
6. Thread library in C++.
7. Demonstrate OO programming concepts through the use of Unreal Game Engine.

EXPECTED LEARNING OUTCOMES

No.	LEARNING OUTCOMES
1.	To demonstrate knowledge and understanding which can serve as the foundation for developing and application of the original ideas.
2.	To apply knowledge, understanding and skills in a broad variety of problems in the field of object oriented programming.
3.	To integrate new knowledge to a successful programming problem solving in programming language C/C++.
4.	To be able to present conclusions and findings to experts and laymen based on the knowledge and experience.
5.	To apply the acquired skills onto further education in this field.

COUPLING OF THE EXPECTED LEARNING OUTCOMES, TEACHING PROCESS ORGANIZATION AND THE EVALUATION OF THE TEACHING OUTCOMES

TEACHING PROCESS ORGANIZATION	ECTS	EXPECTED LEARNING OUTCOMES **	STUDENT ACTIVITY *	EVALUATION METHOD	SCORE	
					min	max

Lecture attendance	1	1-5	Class attendance, discussion, solving the problems individually and in a team	Lists with signatures, observing the activity during the lectures	0	4
Homework	1	1-4	Solving the problems individually	Grading	12	20
Repeated exams	2	1-4	Preparation for the written exam	Grading	19	38
Final exam	2	1-4	Revising	Oral exam	19	38
TOTAL	6				50	100

Teaching methods and student assessment. Lectures will contain many examples with in-depth explanations. Exercises will be held in specialized computer-based laboratories where students will learn how to program in C/C++. The final exam will be held after the completion of lectures and exercises and it will contain practical tasks each student will have to complete independently.

Can the course be taught in English: Yes

Basic literature:
1. B. Strostrup: The C++ Programming Language (4th Ed.). Pearson Education, 2013.

Recommended literature:
1. Materials on the official web site
2. E. Balagurusamy: Object Oriented Programming with C++ (6th Ed.), McGraw Hill Education, 2013.
3. R. Lafore: Object-Oriented Programming in C++ (4th Ed.), Sams Publishing, 2002.
4. J. Šribar, B. Motik: Demistificirani C++, Element, 2010. (3. prošireno izdanje).