

1.	Course Title	Object oriented programming
2.	Code	F18L1S016
3.	Study program	Software engineering and information systems
4.	Study Program Organizer	Faculty of Computer Science and Engineering
5.	Degree (first, second, third cycle)	first cycle
6.	Academic year / semester 1 / summer / mandatory	7. ECTS credits 6
8.	Teacher	full professor Dejan Gjorgjevikj, full professor Ana Madevska Bogdanova, associate professor Nevena Ackovska, associate professor Ivica Dimitrovski, associate professor Ivan Chorbev, associate professor Gjorgji Madzharov, assistant professor Hristina Mihajloska
9.	Course enrollment prerequisites	
10.	<p>Course program goals (competencies):</p> <p>The goal of the course is to acquaint the student with the basic concepts of object-oriented programming. Therefore, the concepts of classes and objects will be introduced, encapsulation, inheritance and polymorphism. The students will be introduced to the concept of hierarchy of classes. Comparisons will be made of the implementation of the object-oriented concepts in different programming languages. After the completion of the course, the student will understand the principles of object-oriented programming and will be able to develop programs based on these concepts and principles.</p>	
11.	<p>Course program content:</p> <p>Introduction to object-oriented programming. Characteristics of OO programming languages. Terminology (objects, classes, methods, encapsulation, abstraction, inheritance, polymorphism). Implementation of methods. Constructors. Destructors. Constructor overloading. Objects as arguments. Embedding objects. Inheritance. Multiple inheritances. Conversions among basic and derived classes. Overloading and redefining functions in derived classes. Polymorphism. Abstract classes. Interfaces. Exceptions.</p>	
12.	<p>Learning methods:</p> <p>Lectures using presentations, interactive lectures, exercises (using equipment and software packages), teamwork, case studies, invited guest lecturers, independent preparation and defense of a project assignment and seminar work.</p>	
13.	Total available time	6 ECTS x 30 hours = 180 hours
14.	Distribution of the available time	30 + 45 + 15 + 15 + 75 = 180 hours

15.	Teaching activity forms	15.1.	Lectures – theoretical teaching	30 hours		
		15.2.	Exercises (laboratory, auditory), seminar papers, teamwork	45 hours		
16.	Other activity forms	16.1.	Project Tasks	15 hours		
		16.2.	Independent Learning Tasks	15 hours		
		16.3.	Home learning	75 hours		
17.	Assessment methodology					
	17.1.	Tests			10 points	
	17.2.	Seminar paper/project (presentation: written and oral)			10 points	
	17.3.	Activity and learning			10 points	
	17.4.	Final exam			70 points	
18.	Assessment criteria (points/grade)	up to 50 points		5 (five) (F)		
		51 to 60 points		6 (six) (E)		
		61 to 70 points		7 (seven) (D)		
		71 to 80 points		8 (eight) (C)		
		81 to 90 points		9 (nine) (B)		
		91 to 100 points		10 (ten) (A)		
19.	Course completion and final exam requirements	Realized activities 15.1 and 15.2				
20.	Teaching Language	Macedonian and English				
21.	Teaching quality evaluation method	Internal evaluation mechanisms and questionnaires				
22.	Course Material					
	22.1.	Mandatory course material				
		No	Author	Title	Publisher	Year
		1	Deitel, Deitel	How to program in Java, 8-th edition	Prentice Hall	2010
		2	Stroustrup B	The C++ Programming Language	AddisonWesley	1997
		3	Bruce Eckel	Thinking in Java, 4th edition	MindView	2006
	22.2.	Additional course material				
		No.	Author	Title	Publisher	Year

