

## Specification for the book of courses

<b>Study program</b>		Electrical Engineering and Computer Science		
<b>Module</b>		Control Systems		
<b>Type and level of studies</b>		Undergraduate Academic Studies		
<b>The name of the course</b>		Fundamentals of Object Oriented Programming		
<b>Lecturer (for lectures)</b>		Janković S. Dragan, Stojković R. Suzana		
<b>Lecturer/associate (for exercises)</b>		Rajković J. Petar, Mihajlović T. Vladan		
<b>Lecturer/associate (for OFE)</b>		Marković M. Ivica, Veljanovski D. Aleksandar		
<b>Number of ECTS</b>		6	<b>Course status (obligatory/elective)</b>	Obligatory
<b>Prerequisites</b>				
<b>Course objectives</b>		Understanding the principles of object oriented paradigm in programming. Learning C++ as a representative object oriented language.		
<b>Course outcomes</b>		Students should fully understand concepts of object oriented programming and be able to develop object oriented programs using programming language C++.		
<b>Course outline</b>				
<b>Theoretical teaching</b>		<p>Programming techniques overview. Problem definition process. Classes. Objects. Using classes in problem solving. Class definition. Class member access. Scope. The separation of interface and implementation. Inline functions. Constructors. Destructors. The call order of constructors and destructors. Copy constructors. Friend classes. Friend functions. Operator overloading. Side effects and connections between operators. Choosing return values for operators. Implementation, inheritance, specialization, generalization. Definition of inherited class. Access modifiers. Types of the inheritance. Constructors and destructors of inherited classes. Pointers and references. Polymorphism. Virtual functions. Pure virtual functions. Abstract classes. Virtual destructors. Arrays and derived classes. Multiple inheritance. Constructors and destructors in multiple inheritance. Multiple derived objects. Virtual base classes. generic mechanism - templates. Template functions. Template classes. Exception handling. Exception throwing. Exception catching. Uncaught exceptions. Input and output streams. Standard streams. Input stream classes. Input stream objects construction. Input stream operations. Extraction operator overloading. Output streams. Stream insertion operator. Output formatting. Output stream operations. Stream insertion operator overloading. Standard library. Namespaces. Container classes (vectors, lists, stack, queues, maps, sets, ...). Iterators, Algorithms, class String</p>		
<b>Practical teaching (exercises, OFE, study and research work)</b>		<p>OO Programming using C++ programming language. Class definition in C++, attributes and methods. Lab: inline and constant functions. Lab: Static members. Constructors and destructors. Lab: Implementing constructors and destructors. Lab: Friend functions and classes. Design lab: Classes, constructors, destructors. Operators. Lab: Operators as class members. Friend operators. Design lab: Operators. Inheritance. Lab: Methods overloading. Lab: Virtual methods. Design lab: Inheritance. Pure virtual methods and abstract classes. Lab: Multiple inheritance. Lab: Virtual base classes. Template classes and template methods. Lab: template class development. Design lab: Template classes. Input and output in C++. Lab: Using standard streams and textual files. Lab: Using binary streams Exception handling. Lab: Exception throwing. Lab: Exception catching. Design lab: using streams and exception handling</p>		
<b>Textbooks/references</b>				
1	M. Stankovic, S. Stojkovic, M.Radmanovic, I. Petkovic, Object oriented languages C++ and Java, Faculty of Electronic Engineering Nis, 2005 (in Serbian)			
2	Laszlo Kraus: Programming language C++, Akademska misao, Beograd, 2007 (in Serbian)			
3	Lectures and exercises as power point presentations			
4				
5				
<b>Number of classes of active education per week during semester/trimester/year</b>				
<b>Lectures</b>	<b>Exercises</b>	<b>OFE</b>	<b>Study and research work</b>	<b>Other classes</b>
2	2	1	0	0
<b>Teaching methods</b>		Lectures, Exercises, Laboratory Exercises		
<b>Grade (maximum number of points 100)</b>				
<b>Pre-exam duties</b>		<b>Points</b>	<b>Final exam</b>	<b>Points</b>
<b>Activity during lectures</b>		5	<b>Written exam</b>	30
<b>Exercises</b>		15	<b>Oral exam</b>	30
<b>Colloquia</b>		20		

Projects			