

STUDY MODULE DESCRIPTION FORM		
Name of the module/subject An introduction to programming		Code 1010331511010334957
Field of study Information Engineering	Profile of study (general academic, practical) general academic	Year /Semester 1 / 1
Elective path/specialty -	Subject offered in: Polish	Course (compulsory, elective) obligatory
Cycle of study: First-cycle studies	Form of study (full-time, part-time) full-time	
No. of hours Lecture: 30 Classes: - Laboratory: 30 Project/seminars: -		No. of credits 6
Status of the course in the study program (Basic, major, other) major		(university-wide, from another field) from field
Education areas and fields of science and art technical sciences		ECTS distribution (number and %) 6 100%
Responsible for subject / lecturer: dr Jerzy Bartoszek email: jerzy.bartoszek@put.poznan.pl tel. 61 665-3713, 61 665-2378 Wydział Elektryczny ul. Piotrowo 3A 60-965 Poznań		
Prerequisites in terms of knowledge, skills and social competencies:		
1	Knowledge	Student has a basic knowledge resulting from the high school
2	Skills	Student is able to meet the challenges arising from the high school.
3	Social competencies	Student has social skills resulting from the high school.
Assumptions and objectives of the course: Basic programming styles and programming concepts with examples of programs in C++/C.		
Study outcomes and reference to the educational results for a field of study		
Knowledge: 1. Student has structured and theoretically founded knowledge of the core. software design, implementation of algorithms, programming paradigms and styles, methods of verifying the correctness of programs, formal languages??, compilers, platforms. - [K_W05]		
Skills: 1. Student is able to use programming environments and platforms to write, perform and test simple programs coded in imperative programming languages??. - [K_U10] 2. Student is able to construct algorithms using basic algorithmic techniques and analyze their complexit. - [K_U09]		
Social competencies: 1. Student is aware of the importance of the accurate completion of the project, notational standards, respect for linguistic correctness and timely submissions. - [K_K07]		
Assessment methods of study outcomes		
Lectures: written tests, pass criterion of 50% points. Laboratory: exercises tests and laboratory reports.		
Course description		

<p>Lectures: Introduction: the structure of simple programs, selected data types, arithmetical and logical operators, expressions, assignments, conditionals, loops, simple I/O statements, namespaces. An introduction to functions. Dynamic and static arrays. References. Structures and operator overloading. Text and binary files. Header files. Dynamic data structures. Selected elements of C. Course update 2017: Pointers and dynamic memory allocation: RAll, smart pointers, make_unique, make_shared. More about functions and their parameters: function overloading, passing arguments, templates, lambdas.</p> <p>Laboratory: An introduction: main, int, std::string, arithmetic operators, if/else, cin/cout, debugger. Simple types and Loops. SVN. Funtions. Dynamic and static arrays. References: std::vector, std::array, for_each, auto. Structures. Text and binary files: std::fstream, reinterpret_cast. Header files. Namespaces. Function and operator overloading. Pointers and dynamic memory allocation: RAll, smart pointers, make_unique, make_shared. Lambdas. Templates. How to read C programs?: printf, scanf, malloc, free, static and dynamic arrays.</p> <p>Teaching methods: lectures - with multimedia presentation, additional topics included in Moodle course laboratory - with multimedia presentation, additional topics included in Moodle course, used tools enable students to perform tasks at home</p>		
<p>Basic bibliography:</p> <ol style="list-style-type: none"> 1. Grębosz J., Symfonia C++ standard, Programowanie w języku C++ orientowane obiektowo, T.1 i 2 2. Stroustrup B., Programming - Principles and Practice Using C++ 3. http://en.cppreference.com/w/ 4. https://isocpp.org/faq 5. https://msdn.microsoft.com/en-us/library/3bstk3k5.aspx 6. http://www.cplusplus.com/ 		
<p>Additional bibliography:</p> <ol style="list-style-type: none"> 1. Banachowski L., Kreczmar A., Rytter W., Analysis of Algorithms and Data Structures, Addison Wesley, 1991 		
<p>Result of average student's workload</p>		
<p>Activity</p>		<p>Time (working hours)</p>
1. participation in lectures		30
2. participations in labs.		30
3. exam, consultation		10
4. preparation for labs., reports		45
5. preparation for tests and exam		35
<p>Student's workload</p>		
<p>Source of workload</p>	<p>hours</p>	<p>ECTS</p>
Total workload	150	6
Contact hours	75	3
Practical activities	75	3