



Subject card

Subject name and code	Basics of computer programming, PG_00045290						
Field of study	Data Engineering						
Date of commencement of studies	October 2022	Academic year of realisation of subject			2022/2023		
Education level	first-cycle studies	Subject group			Obligatory subject group in the field of study Subject group related to scientific research in the field of study		
Mode of study	Full-time studies	Mode of delivery			at the university		
Year of study	1	Language of instruction			English		
Semester of study	1	ECTS credits			5.0		
Learning profile	general academic profile	Assessment form			assessment		
Conducting unit	Department of Algorithms and Systems Modelling -> Faculty of Electronics, Telecommunications and Informatics						
Name and surname of lecturer (lecturers)	Subject supervisor	prof. dr hab. inż. Dariusz Dereniowski					
	Teachers	prof. dr hab. inż. Dariusz Dereniowski dr hab. inż. Robert Janczewski mgr inż. Robert Ostrowski mgr inż. Andrzej Jastrzębski dr inż. Tytus Pikiś					
Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	15.0	20.0	0.0	65
	E-learning hours included: 0.0						
	Podstawy Programowania 2022/23 (Informatyka & Inżynieria Danych) - Moodle ID: 23902 https://enauczanie.pg.edu.pl/moodle/course/view.php?id=23902						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM		
	Number of study hours	65	10.0	50.0	125		
Subject objectives	The aim of the course is an introduction to computer programming, whose main goal is to teach student solving programming tasks and writing programs in C/C++.						
Learning outcomes	Course outcome	Subject outcome			Method of verification		
	[K6_U01] programs in procedural, object, functional and logic programming languages, codes programs at the processor instruction level, runs and tests programs.	Student is able to code in procedural programming language, runs and tests programs.			[SU1] Assessment of task fulfilment		
	[K6_W05] Knows and understands programming models and evolution of related languages. Knows the methods of analysing and designing information systems and the modeling languages used in them, as well as the basic object-oriented programming platforms.	Student knows and understands selected programming models and corresponding programming languages. Student learns one of the object oriented programming platforms.			[SW1] Assessment of factual knowledge		

Subject contents	<p>LECTURES Introduction. Programming languages, alphabet, syntax and semantics. Translation. Classification of data types. Integer and floating point types. Arithmetic expressions and operators. Standard mathematical functions. Character type. Casting. Logical type. Logical operators and expressions. Basics of input/output processing. Conditional statements (if, switch) and conditional expression. Iteration statements (for, while, do-while). Nested iterations. Defining types. Enumerated type. Constants. One- and multi-dimensional arrays. Null-terminated strings. Scope and lifetime of variables. Functions. Side effect. Passing parameters to functions. Pointer type. Pointer arithmetic. Pointers for inter-function communication. Dynamic memory allocation. Basic dynamic data structures. Records (structures). Data structures using records and their applications. Applications of dynamic data structures (stacks, queues). Introduction to object oriented programming. Defining classes. Applications of object oriented paradigm. Input/output streams classes. Input/output formatting. File processing. Applications of recurrence (e.g. divide and conquer, greediness, dynamic programming).</p> <p>LABORATORIES Solving simple programming tasks according on knowledge provided in lectures and based on provided manual.</p> <p>PROJECT Independent solving programming tasks. Student has access to dedicated tutors.</p>														
Prerequisites and co-requisites	No requirements														
Assessment methods and criteria	<table border="1"> <thead> <tr> <th data-bbox="454 698 796 730">Subject passing criteria</th> <th data-bbox="799 698 1141 730">Passing threshold</th> <th data-bbox="1144 698 1482 730">Percentage of the final grade</th> </tr> </thead> <tbody> <tr> <td data-bbox="454 734 796 766">written test</td> <td data-bbox="799 734 1141 766">50.0%</td> <td data-bbox="1144 734 1482 766">40.0%</td> </tr> <tr> <td data-bbox="454 770 796 801">laboratories</td> <td data-bbox="799 770 1141 801">50.0%</td> <td data-bbox="1144 770 1482 801">30.0%</td> </tr> <tr> <td data-bbox="454 806 796 837">project</td> <td data-bbox="799 806 1141 837">50.0%</td> <td data-bbox="1144 806 1482 837">30.0%</td> </tr> </tbody> </table>			Subject passing criteria	Passing threshold	Percentage of the final grade	written test	50.0%	40.0%	laboratories	50.0%	30.0%	project	50.0%	30.0%
Subject passing criteria	Passing threshold	Percentage of the final grade													
written test	50.0%	40.0%													
laboratories	50.0%	30.0%													
project	50.0%	30.0%													
Recommended reading	<p>Basic literature</p> <p>Supplementary literature</p> <p>eResources addresses</p>	<p>1. Programming – lecture slides, 2013 (available on course website).</p> <p>2. Materials for laboratories (2013) (available on course website).</p> <p>3. Grębosz Jerzy, Symfonia C++ Standard (vol. 1 and 2), Edition 2000, Cracow 2008.</p> <p>-</p>													
Example issues/ example questions/ tasks being completed	<p>Writing a program that fulfills a given specification.</p> <p>Analysis of a behavior of a given code.</p>														
Work placement	Not applicable														