

Subject card

Subject name and code	Introduction to Programming, PG_00189367						
Field of study	Mathematical Modeling and Data Analysis						
Date of commencement of studies	October 2026		Academic year of realisation of subject		2026/2027		
Education level	Bachelor's studies		Subject group		Obligatory subject group in the field of study		
Mode of study	full-time studies		Mode of delivery		at the university		
Year of study	1		Language of instruction		Polish		
Semester of study	1		ECTS credits		4.0		
Learning profile	academic		Assessment form		credit		
Conducting unit	Institute of Mathematics -> Faculty of Mathematics, Physics and Informatics -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		dr Iwona Krzyżanowska				
	Teachers		dr Iwona Krzyżanowska				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	0.0	0.0	60.0	0.0	0.0	60
	E-learning hours included: 0.0						
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	60		3.0		37.0	100
Subject objectives	Students acquire the ability to algorithmize problems and implement them in a high-level, general-purpose programming language, e.g. Python, based on the classic elements of this language. Developing the ability to control and verify one's own and other people's reasoning and source code. Preparation for in-depth intellectual, literary and technical exploration of IT problems for which simple tools are not sufficient.						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[MMiADL3_U07] is able to recognize problems, including practical issues, that can be solved algorithmically; is able to make specification for such a problem		The student can create an algorithm for a new problem, justify its correctness, and implement it in a learned programming language.		[SU5] implementation of a problem task		
	[MMiADL3_U08] is able to create and analyze an algorithm consistent with the specification and write it in a selected programming language, as well as compile, run and test a computer program written by himself/herself		The student is able to formulate an algorithm for a problem, justify its correctness, and implement this algorithm in a learned programming language, selecting appropriate data types and control structures. The student can design the structure of a program by extracting subprograms (functions). The student can compile, run, and test a computer program written independently in a high-level language, such as Python.		[SU5] implementation of a problem task		
	[MMiADL3_U12] is able to work in a team, plan and organize such work		The student is able to plan, divide, and carry out the work on a programming project.		[SU5] implementation of a problem task [SU8] observation of student's independent or team work		

Subject contents	1. Safety and hygiene rules in the computer laboratory 2. Preparing the environment. 3. Types of variables (lists, dictionaries, tuples, sets). 4. Creating simple scripts. 5. Conditional if statement. 6. Pseudo-random numbers. 7. For, while loops. 8. Working with a file, writing, reading, modifying. 9. Testing and error handling. 10. Writing your own functions. 11. GUI graphical user interface, e.g. tkinter module, graphics.py 12. Introduction to the classes.		
Prerequisites and co-requisites			
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	observation of the student's attitude	51.0%	0.0%
	tests and/or project	51.0%	100.0%
Recommended reading	Basic literature	1. https://docs.python.org/3/ 2. Lutz M., Python. Wprowadzenie.	
	Supplementary literature	• Heinold B., A Practical Introduction to Python Programming • McKinney W., Python w analizie danych. Przetwarzanie danych za pomocą pakietów Pandas i NumPy oraz środowiska IPython	
	eResources addresses		
Example issues/ example questions/ tasks being completed	none		
Work placement	Not applicable		

Document generated electronically. Does not require a seal or signature.