

Subject card

Subject name and code	Introduction to Programming, PG_00167091						
Field of study	Wstęp do programowania						
Date of commencement of studies	October 2025		Academic year of realisation of subject		2025/2026		
Education level	Bachelor's studies		Subject group		Obligatory subject group in the field of study Optional subject group		
Mode of study	full-time studies		Mode of delivery		at the university		
Year of study	1		Language of instruction		Polish polish		
Semester of study	1		ECTS credits		5.0		
Learning profile	academic		Assessment form		credit		
Conducting unit	Faculty of Mathematics, Physics and Informatics -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		dr Sławomir Werbowy				
	Teachers		dr Sławomir Werbowy				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	60.0	0.0	0.0	75
	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	75		0.0		60.0	135
Subject objectives	Introduction to programming and elements of algorithm theory and data structures.						

Learning outcomes	Course outcome	Subject outcome	Method of verification
	[BJORL3_U04] Can use mathematical and computer apparatus to analyze and solve problems in radiological protection and nuclear safety	- The student is able to precisely formulate a computational problem for further analysis. - The student can implement the formulated problem in a selected programming language. - The student can analyze and interpret the results to deepen the understanding of the studied subject.	[SU5] realizacja zadania problemowego [SU6] demonstracja umiejętności praktycznych [SU8] obserwacja samodzielnej lub zespołowej pracy studenta
	[BJORL3_W02] Understands the role of physical and chemical experimentation, mathematical theoretical models approximating reality, and computer simulations in scientific research methodology; is aware of technological, apparatus, and methodological limitations in scientific research	Student: - has knowledge of structured and object-oriented programming in a selected high-level language (Python) - knows data types, control statements, arithmetic and logical operators in Python; - understands object-oriented programming concepts and can apply them in practical tasks; - knows and can use computational packages and tools for symbolic calculations; - knows and can use user software packages for data analysis and results presentation.	[SW5] realizacja zadania problemowego
	[BJORL3_U02] Has the ability to perform measurements of basic quantities used in physics and chemistry; can develop, describe and present the results of simple experiments and computer simulations; can perform quantitative analyses and formulate qualitative conclusions on this basis; can estimate measurement uncertainties	The student can: formulate a simple numerical algorithm to solve a given problem. - write and run a computer program in Python, processing numerical and text data. - apply structured and object-oriented programming approaches to develop programs ready for testing and documentation. - write a program that reads from and writes to alphanumeric files. -test a program, identify and correct errors, and prepare documentation describing the program's functionality.	[SU5] realizacja zadania problemowego [SU6] demonstracja umiejętności praktycznych [SU8] obserwacja samodzielnej lub zespołowej pracy studenta
Subject contents	1. Introduction to Python and the Spyder IDE. 2.Data types, arithmetic and logical operators, conditional and control statements. 3. Complex data types: lists, tuples, dictionaries. 4. Procedural programming: functions, modules, basic work with packages. 5. Reading from and writing to alphanumeric files. 6. Elements of object-oriented programming: classes, objects, methods. 7. Basics of computational and data visualization packages.		
Prerequisites and co-requisites			
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	not applicable	0.0%	20.0%
	not applicable	51.0%	80.0%
Recommended reading	Basic literature	A.1. Used during classes: Instructions and materials provided by the instructor. A.2. Recommended for self-study: Eric Matthes, Python Crash Course, 3rd Edition, No Starch Press, 2022. Al Sweigart, Automate the Boring Stuff with Python, No Starch Press, 2019. Mark Reed, Python Programming for Beginners, Independently Published, 2022. Python Documentation: https://docs.python.org/3/ Online materials on computational and data visualization packages, e.g., NumPy and Matplotlib.	
	Supplementary literature	R. Johansson, Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib. Apress 2024	
	eResources addresses		
Example issues/ example questions/ tasks being completed	Write a Python program that: 1. Reads experimental measurement data from a CSV file (e.g., time, temperature, voltage). 2. Performs basic data analysis: calculates mean values, standard deviations, minimum and maximum values for each column. 3. Creates plots showing relationships between the data (e.g., temperature vs. voltage).		
Work placement	Not applicable		

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