

Subject card

Subject name and code	Introduction to Python programming, PG_00054863						
Field of study	Chemistry						
Date of commencement of studies	October 2026		Academic year of realisation of subject		2026/2027		
Education level	Master'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		English		
Semester of study	1		ECTS credits		2.0		
Learning profile	academic		Assessment form		exam		
Conducting unit	Faculty of Chemistry -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		prof. dr hab. Cezary Czaplewski				
	Teachers						
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	0.0	0.0	0.0	15
	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	15		2.0		33.0	50
Subject objectives	Ability to properly design basic algorithms, introduction to programming in Python						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[CHEMMU2_W06] Applies mathematics to the extent necessary to understand, describe and model chemical processes of medium complexity.		The student defines the concept of the algorithm. Characterizes the most important classes of algorithms.		[SW4] test/exam - oral or written [SW2] presentation/project/paper/report		
	[CHEMMU2_U11] Communicates in English in accordance with the requirements specified for level B2 of the Common European Framework of Reference for Languages.		A student can use specialist terminology associated with algorithms and the Python programming language.		[SU2] presentation/project/paper/report [SU5] implementation of a problem task		
	[CHEMMU2_K06] Undertakes research tasks consciously and responsibly, understanding the social aspects of the practical application of the acquired knowledge and skills and the responsibility related to it.		Learns the principles of working safely, responsibly, and efficiently using the workstations connected to the Internet. Develops the responsibility for his/her personal account on the workstation.		[SK8] observation of student's independent or team work		
	[CHEMMU2_W05] Has extended knowledge in the field of the specialisation studied.		Names and describes the types and the data structures of the Python programming language. Differentiates the Python control instructions.		[SW4] test/exam - oral or written [SW2] presentation/project/paper/report		
	[CHEMMU2_K01] Knows the limitations of her/his own knowledge; understands the need for further education and can inspire other people to do so.		Develops the ability of working in a team.		[SK5] implementation of a problem task [SK8] observation of student's independent or team work		

Subject contents	Algorithms and data structures in Python. Review of the most important data structures that are helpful in programming, objects, expressions, and numerical types. Functions and scoping. Testing and debugging. Handling exceptions. Classes and object-oriented programming. The most important programming algorithms: approximate estimation of the complexity of algorithms, brute-force algorithms, divide and conquer algorithms, recursion. Versions of Python language, useful Python libraries: NumPy, Scipy.		
Prerequisites and co-requisites			
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	test exam	51.0%	100.0%
Recommended reading	Basic literature	Mark Lutz, Learning Python, O'Reilly Media, Inc, USA, 2013 John V. Guttag Introduction to Computation and Programming Using Python, MIT Press 2013	
	Supplementary literature	Alberto Boschetti, Luca Massaron, Python Data Science Essentials, PublishDrive 2018 Svein Linge, Hans Petter Langtangen Programming for Computations Python, Springer 2020 Joakim Sundnes Introduction to Scientific Programming with Python, Springer 2020	
	eResources addresses		
Example issues/ example questions/ tasks being completed			
Work placement	Not applicable		

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