

Fundamentals of programming - course description

General information	
Course name	Fundamentals of programming
Course ID	13.2-WF-FizP-FP-S21
Faculty	Faculty of Physics and Astronomy
Field of study	Physics
Education profile	academic
Level of studies	First-cycle studies leading to Bachelor's degree
Beginning semester	winter term 2022/2023

Course information	
Semester	1
ECTS credits to win	5
Course type	obligatory
Teaching language	english
Author of syllabus	<ul style="list-style-type: none">dr Marcin Kośmider

Classes forms					
The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Laboratory	60	4	-	-	Credit with grade

Aim of the course

The aim of the course is to learn the basics of programming and the ability to use the acquired knowledge to solve a variety of problems, with particular emphasis on problems related to exact sciences. This approach to programming requires understanding not only the syntax of a programming language, but also the basics of algorithmics, software development phases, coding standards, the ability to work with documentation, and the analysis and reduction of a complex problem to a series of elementary problems. The basics of programming are also the necessary foundation for understanding other computer subjects such as numerical methods, object-oriented programming or modeling and computer simulations.

Prerequisites

Basic computer skills

Scope

1. A brief history and characteristics of the Python language
2. Work environment, naming conventions
3. Data types, variables, substitution operator, logical and mathematical operators
4. Conditional statement, conditional operator
5. Loops
6. Sequential data types: strings, lists, tuples, dictionaries, sets
7. Functions
8. Standard modules and packages, own modules
9. Exceptions
10. File operations
11. The concept of class and object, methods
12. Using numpy for calculations and simulations
13. Matplotlib - introduction to data visualization

Teaching methods

Discussion, group work, work with documentation, brainstorming, lecture, presentation

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
---------------------	-----------------	-------------------------	----------------

Outcome description	Outcome symbols	Methods of verification	The class form
The student is aware of the existence of Open Source software that is a professional alternative to commercial software. The student is aware of the speed of changes in the IT industry and the related need to constantly improve their competences. The student is able to create and present a report on the entrusted project.	<ul style="list-style-type: none"> • K1A_W09 • K1A_U08 • K1A_K01 • K1A_K04 • K1A_K06 	<ul style="list-style-type: none"> • a discussion 	<ul style="list-style-type: none"> • Laboratory
The student is able to independently search and use tools and information helpful in solving a given problem.	<ul style="list-style-type: none"> • K1A_W09 • K1A_U07 	<ul style="list-style-type: none"> • a quiz 	<ul style="list-style-type: none"> • Laboratory
The student is able to define and explain the problem posed by breaking it down into elementary problems and presenting methods (algorithms) for the optimal solution to the problem.	<ul style="list-style-type: none"> • K1A_W03 • K1A_U03 • K1A_U05 	<ul style="list-style-type: none"> • a discussion • a quiz 	<ul style="list-style-type: none"> • Laboratory
The student knows data types, control instructions, functions, can work with static and dynamic arrays and IO streams. Can use the knowledge and available tools to present a solution to a problem (in particular in the field of physics and related fields) in the form of source code	<ul style="list-style-type: none"> • K1A_W04 • K1A_W09 • K1A_U04 • K1A_U05 	<ul style="list-style-type: none"> • a quiz 	<ul style="list-style-type: none"> • Laboratory
The student knows the regulations and health and safety rules in force in the computer lab.	<ul style="list-style-type: none"> • K1A_W06 	<ul style="list-style-type: none"> • a discussion 	<ul style="list-style-type: none"> • Laboratory

Assignment conditions

The final grade consists of: 10% is the average of active participation in the classroom, 40% is the average of tests / tests during the semester, 50% is the grade of the final project.

Recommended reading

1. "Python. Wprowadzenie. Wydanie IV", M.Lutz, Helion
2. "Python dla każdego. Podstawy programowania. Wydanie III", M.Dawson, Helion
3. "Automatyzacja nudnych zadań z Pythonem. Nauka programowania", A. Sweigart

Further reading

1. "The Complete Python Course For Beginners" youtube na kanale Tech with Tim (<https://www.youtube.com/watch?v=sxTmJE4k0ho>)

Notes

Modified by dr Marcin Kośmider (last modification: 04-04-2022 20:43)

Generated automatically from SylabUZ computer system