Bialystok University of Technology Faculty of Electrical Engineering Department of Electrotechnics, Power Electronics and Electrical Power Engineering

Instruction for a specialist workshop on

Python Programming 1

Subject code: CP1S02005E

(Full-Time Studies)

PYTHON - ELEMENTS OF OBJECT-ORIENTED PROGRAMMING

Instruction Number

PP_09_EN

Author: Jarosław Forenc, PhD

Bialystok 2025

Contents

1.	Workstation Description				
	1.1.	Equipment used	3		
	1.2.	Software	3		
2.	Exercises Procedure				
3.	Literature				
4.	Healt	h and Safety Requirements	Safety Requirements		

Teaching Materials for Students of the Faculty of Electrical Engineering at BUT.

[©] Faculty of Electrical Engineering, Bialystok University of Technology, 2025 (version 1.0)

All rights reserved. No part of this publication may be copied or reproduced in any form or by any means without the permission of the copyright holder.

1. Workstation Description

1.1. Equipment used

A PC-class computer with the Microsoft Windows 10 operating system is used during classes.

1.2. Software

The computers are equipped with the Visual Studio Code 1.86 (or newer) source code editor along with the appropriate extension (Python extension for Visual Studio Code).

2. Exercises Procedure

In the specialist workshop, selected tasks assigned by the instructor must be completed. Different groups may perform different tasks.

- Define a class named **Student** that stores information about a student (first name, last name, student ID number, year of study). Create a constructor that initializes all attributes of the student. Add a method to the class that displays the student's information on the screen. Create several objects of the **Student** class and call the display method for each of them. Consider adding other methods, such as updating the student's data.
- Define a class Function that stores a quadratic function: ax² + bx + c. The class should contain three attributes: a, b, and c, initialized in the constructor. Define a method that returns the roots (zeros) of the stored function. Take into account the cases where the coefficients a = 0, b = 0, or c = 0.
- 3. Write a program that includes a class **TComplex** designed to perform operations on complex numbers. The class should have two attributes: **Re** the real part of the complex number, and **Im** the imaginary part. Define methods to perform the following operations:

- a) assign values to the class attributes (constructor);
- b) display the complex number in the format: **Re ± Im j**, e.g., **5 3j**, **0 + 1j**;
- c) return the real part of the complex number;
- d) return the imaginary part of the complex number;
- e) calculate and return the modulus (absolute value) of the complex number;
- f) calculate and return the argument of the complex number in radians;
- g) calculate and return the argument of the complex number in degrees;
- h) calculate and return the complex conjugate;
- i) calculate and return the sum of two complex numbers;
- j) calculate and return the difference between two complex numbers.

Call the defined functions for the following complex numbers:

z1 = 2 + 4j z2 = 6 - 8j

Verify the correctness of the results with the following table:

Operation	Result	Operation	Result
z1 + z2	8 - j4	z1 - z2	-4 + j12
z1*	2 - j4	z2*	6 + j8
Abs(z1)	4,472	Abs(z2)	10
ArgRad(z1)	1,107	ArgRad(z2)	-0,927
ArgDeg(z1)	63,435	ArgDeg(z2)	-53,13

Table 1. Correct results for Task 3

- Define a class **TPoint** that represents a point in a rectangular coordinate system. The class should store the **X** and **Y** coordinates of the point. Define methods to perform the following operations:
 - a) assign values to the class attributes (constructor);
 - b) display the coordinates in the form **[X, Y]**;

- c) return the distance from the origin (0, 0);
- d) return the distance between two **TPoint** objects;
- e) move the point by **dx** in the X direction and **dy** in the Y direction;
- f) calculate and return the number of the quadrant in which the point lies (if the point lies on the X or Y axis, return **0**).

Demonstrate the use of all defined methods in the program.

- 5. Write a program with two classes: Point and Circle. The Point class represents a point in a rectangular coordinate system and contains two attributes: x and y. The Circle class inherits from Point. It represents a circle's position in the coordinate system and includes an additional attribute R (radius). Define constructors and methods to display the values of the attributes in both classes. In the Point class, define a method to update the x and y coordinates, and a method to move the point by dx and dy. In the Circle class, define a method to calculate the area of the circle.
- 6. Write a program with two classes: Vehicle and PassengerCar. The Vehicle class should contain the following attributes: brand, model, color, year of production, and mileage (must not be negative). The PassengerCar class should contain the following attributes: weight (should be between 1 and 3.5 tons), engine capacity (should be between 0.8 and 3.0 cm³), and number of seats. The PassengerCar class should inherit from Vehicle. In both classes, create a constructor that reads the data from the user. Add methods to display information about the vehicles.

3. Literature

- [1] Ramalho L., Fluent Python: clear, concise, and effective programming. Sebastopol, O'Reilly, 2022.
- [2] Matthes E., Python Crash Course, San Francisco, CA, No Starch Press, 2019.

- [3] Sweigart A., Automate the Boring Stuff with Python, San Francisco, CA, No Starch Press, 2020.
- [4] Lutz M., Learning Python, Sebastopol, CA, O'Reilly Media, 2013.
- [5] <u>https://www.python.org/doc/</u> Python, documentation.

4. Health and Safety Requirements

To begin the practical part of the exercise, it is mandatory to familiarize yourself with the health and safety instructions and fire safety guidelines and to adhere to the rules contained therein.

During laboratory sessions, the following rules must be observed:

- Verify that the devices available at the laboratory workstation are complete and show no signs of physical damage.
- If possible, adjust the workstation conditions to suit individual ergonomic needs. Position the computer monitor to ensure constant and comfortable visibility for all team members.
- Check the correctness of device connections.
- The computer may only be turned on with the instructor's permission.
- Eating and drinking are prohibited while working with the computer.
- Upon completion of work, log out before leaving the workstation. The operating system may only be shut down upon explicit instruction from the instructor.
- Making any modifications, switching components, or replacing elements of the workstation is strictly prohibited.
- Changing the computer's configuration, including the operating system and software, is not allowed unless it is part of the class program and performed under the instructor's supervision.
- In the event of a power failure, immediately turn off all devices.
- Any missing equipment or malfunctions must be reported to the instructor.

- It is forbidden to operate, manipulate, or use devices not included in the current exercise.
- In case of electric shock, immediately disconnect the workstation from the power supply. Do not touch the affected person before the power is turned off.