COURSE DESCRIPTION CARD - SYLLABUS

Course name
Computer programming 1 [N1IZarz1>PROG1]

Course

Field of study
Engineering Management

Year/Semester
1/2

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
part-time

Requirements
compulsory

Number of hours

Lecture
10

Laboratory classes
12

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

3,00

Coordinators

dr inż. Zbigniew Włodarczak
zbigniew.wlodarczak@put.poznan.pl

Lecturers

Prerequisites

Basic computer skills and knowledge of spreadsheets with macro elements.

Course objective

The aim of the course is to prepare students to independently create simple programs and understand basic algorithmic structures, with particular emphasis on the practical use of C# or Python.

Course-related learning outcomes

Knowledge:
The student describes object-oriented programming language, its properties, and applications [P6S_WG_08].
The student defines event handling procedures in the context of programming [P6S_WG_08].
The student names functions, control statements, operators, and selected data types used in programming [P6S_WG_08].

Skills:
The student plans and conducts experiments, including measurements and computer simulations, to

test programs [P6S_UW_09].
The student performs interpretation of experiment results and draws conclusions from the conducted measurements and computer simulations [P6S_UW_09].
The student prepares and creates functions and control instructions in programs [P6S_UO_01].
The student utilizes declarations, operators, and selected data types in programs [P6S_UW_09, P6S_UO_01].

Social competences:
The student recognizes cause-and-effect relationships in achieving programming goals and prioritizes the importance of alternative tasks in the process of program creation [P6S_KK_02].

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lecture: Knowledge acquired during the lecture is verified by completing one problem-solving task and a final test, pass mark: 50% of points.
Laboratory: During the semester, 2-4 tasks described in the course must be completed. Each task is assessed on a scale of 0-100 points. The point score is converted into a final grade.

Programme content

Object-oriented programming language, object properties, event handling procedures, use and creation of functions, control instructions and use of declarations, operators and selected data types.

Course topics

Lecture: Introduction to programming, procedural programming, structured programming, object-oriented programming.
Laboratory: Object-oriented programming language (C# or Python).

Teaching methods

Lecture: informative lecture, problem-based lecture, conversational lecture, case study method.
Laboratory: laboratory method (experiment), workshop method.

Bibliography

1. Michaelis M., C# 8.0. Essential C# 8.0, Addison-Wesley Microsoft Technology Series, 2020.
2. Jamro M., C# Data Structures and Algorithms: Explore the possibilities of C# for developing a variety of efficient applications

Additional:
1. Price M.J., C# 13 and .NET 9 – Modern Cross-Platform Development Fundamentals: Start building websites and services with ASP.NET Core 9, Blazor, and EF Core 9, Packt Publishing, 2025.

Breakdown of average student's workload

	Hours	ECTS
Total workload	75	3,00
Classes requiring direct contact with the teacher	22	1,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	53	2,00