COURSE DESCRIPTION CARD - SYLLABUS

Course name
Informatics II [S1EiT1>INF3]

Course

Field of study
Electronics and Telecommunications

Year/Semester
2/4

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes
30

Other (e.g. online)
0

Tutorials
0

Projects/seminars
0

Number of credit points

5,00

Coordinators

prof. dr hab. inż. Grzegorz Danilewicz
grzegorz.danilewicz@put.poznan.pl

Lecturers

Prerequisites

Students taking the course should have a basic knowledge of computer design and number systems. They can use high-level programming languages C and C++. Additionally, students can acquire information from literature, databases, and other sources in Polish or English. They should be able to integrate, interpret, and analyze this information to draw conclusions and support their opinions.

Course objective

Introduce students to the core concepts of object-oriented programming, including classes, objects, inheritance, polymorphism, and encapsulation. Develop proficiency in using essential Python libraries for different tasks. Compare and contrast object-oriented and functional programming paradigms through practical Python examples, emphasizing their strengths and weaknesses in different problem-solving contexts.

Course-related learning outcomes

Knowledge:
1. The student has a basic knowledge of development trends in the field of: high-level programming languages included in the .NET platform, organization of modern programming platforms and

integration of programming languages.
2. The student knows high-level programming Python language. (S)he knows the principles of designing computer programs with an extensive graphical interface, creating multithreaded software, cooperation with databases, and the basic mechanisms related to programming network applications using the Python language.

Skills:
1. The student can use the high-level programming Python language; can design and program applications with an extensive graphic interface; knows how to create software implementing basic network protocols and cooperating with basic database suppliers; can design and create multithreaded software.
2. The student can solve typical engineering programming problems with the use of Python language.

Social competences:
1. The student knows the limitations of his own knowledge and skills in the field of modern high-level programming languages, understands the need for further training in programming languages and platforms
2. The student is aware of the need for a professional approach to solving technical problems with the use of modern languages and programming platforms; student is aware of the responsibility taken for the software (s)he develops.
3. The student is aware of the dangers of badly designed and developed software, both for users and devices.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Lectures. Knowledge is checked during two tests. Written tests concern the lectures content - one in the middle of the lectures, the other at the end of the lectures. Each of the tests must be passed with at least a satisfactory grade. A satisfactory grade is issued when the number of points for the test is greater than 50%. The final grade is the arithmetic mean of the test grades. For the test grades and the final grade, a grading scale from 2 (unsatisfactory - negative) to 5 (very good) is used. Rules for converting the final grade:
Average range : grade
0,00 - 2,99 : 2,0
3,00 - 3,24 : 3,0
3,24 - 3, 74 : 3,5
3,75 - 4,24 : 4,0
4,25 - 4,74 : 4,5
4,75 - 5,00 : 5,0
Laboratories. The following components are assessed: student"s knowledge before the exercise, answers to questions during the exercises, written reports on the implementation of the exercises, written test at the end of the semester. For the component grades and the final grade, a grading scale from 2 (unsatisfactory - negative) to 5 (very good) is used.

Programme content

The program content covers:
1. compilation versus code interpretation,
2. structured versus object-oriented programming,
3. Python as a language for research and prototyping, and
4. Python programming basics.

Course topics

1. Introduction to Python programming
2. The Python programming environment
3. Commenting on code
4. Simple data types: integers, floats, booleans
5. Strings: formatting, methods, operations

6. Using libraries and modules
7. Functions: definition, calling, parameters, return values
8. Anonymous functions (lambda)
9. Input and output operations
10. Numerical computations and operators
11. Conditional statements and logical operators
12. Looping constructs
13. Data structures: lists, tuples, dictionaries, sets
14. File handling: text and binary files
15. Exception handling
16. Object-oriented programming: classes, objects, inheritance, polymorphism
17. Creating and using custom modules
18. Data exchange formats (JSON, XML)
19. Data visualization with Python libraries (Matplotlib)
20. Advanced Python topics
Laboratory: Practical Python programming projects

Teaching methods

1. Lectures
a. Traditional lecture
b. Lecture with Jupyter notebook
c. Discussion-based lecture
d. Experiment
e. Case study
f. Software development
2. Exercises
Students will complete practical exercises using Python programming language runtime environments. These exercises will be assigned by the instructor and accompanied by multimedia presentations.

Bibliography

Basic
1. Jesse Liberty "Programowanie C#", Helion 2005
2. Charles R. Severance, Python for Everybody. Exploring Data in Python 3, Charles Severance, University of Michigan
Additional
3. www.python.org

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,00
Classes requiring direct contact with the teacher	70	3,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	55	2,00