Syllabus

1. Programme information

1.1. Institution	THE BUCHAREST UNIVERSITY OF ECONOMIC STUDIES
1.2. Faculty	Business Administration in Foreign Languages
1.3. Departments	Department of Economic Informatics and Cybernetics
1.4. Field of study	Business Administration
1.5. Cycle of studies	Licence
1.6. Education type	Full-time
1.7. Study programme	Business Administration
1.8. Language of study	English
1.9. Academic year	2022-2023

2. Information on the discipline

2.1. Name	**Basics of programming**								
2.2. Code	**22.0153IF1.1-0003**								
2.3. Year of study	**1**	2.4. Semester	**1**	2.5. Type of assessment	**Exam**	2.6. Status of the discipline	**O**	2.7. Number of ECTS credits	**5**

2.8. Leaders	C(C)	**conf.univ.dr. VESPAN Dragos Marcel**	dragos.vespan@ie.ase.ro
	L/P(L/P)	**conf.univ.dr. VESPAN Dragos Marcel**	dragos.vespan@ie.ase.ro

3. Estimated Total Time

3.1. Number of weeks	14.00		
3.2. Number of hours per week	4.00	of which	
		C(C)	2.00
		L/P(L/P)	2.00
3.3. Total hours from curriculum	56.00	of which	
		C(C)	28.00
		L/P(L/P)	28.00
3.4. Total hours of study per semester (ECTS*25)	125.00		
3.5. Total hours of individual study	69.00		

Distribution of time for individual study	
Study by the textbook, lecture notes, bibliography and student's own notes	14.00
Additional documentation in the library, on specialized online platforms and in the field	15.00
Preparation of seminars, labs, assignments, portfolios and essays	28.00
Tutorials	10.00
Examinations	2.00
Other activities	0.00

4. Prerequisites

4.1. of curriculum	
4.2. of competences	Computer operation Use of Office suite of programs (beginner)

5. Conditions

for the C(C)	Class with computer connected to beamer, Internet access. Installed software: MS Office, Anaconda, Flowgorithm
for the L/P(L/P)	Laboratories with computers connected to the internet, with one computer for each student. Installed software: Anaconda, Flowgorithm

6. Acquired specific competences

PREFESSIONAL	C5	Utilization of databases specific to business administration

7. Objectives of the discipline

7.1. General objective	Development of skills and competencies in computational thinking and in problem solving using the computer, through programming in Python language.
7.2. Specific objectives	Learning the principles of computational thinking: decomposition into sub-problems, pattern recognition, abstraction and algorithmic thinking Knwledge of basics of programming in Python: input and output, variables and expressions, control statements, functions, object oriented programming concepts Acquiring skills or self documentation / searching for relevant information to solve problems independently

8. Contents

8.1. C(C)		Teaching/Work methods	Recommendations for students
1	Presentation of the objectives, the subject of the course and the evaluation methods. Introductory elements of computer programming. Examples of algorithms	Presentation with PowerPoint support and practical exemplification	
2	Basic elements of the Python language. Development environments, installation and working modes. Input/output functions	Presentation with PowerPoint support and practical exemplification	
3	Basic data types in Python. Variables, operators and expressions. Internal representation of numerical data.	Presentation with PowerPoint support and practical exemplification	
4	Scientific notation. Data types int and float - fixed point and floating point.	Presentation with PowerPoint support and practical exemplification	
5	Elements of Boolean logic in the decision-making process. Logical Operators in Python.	Presentation with PowerPoint support and practical exemplification	
6	Representation of control flows and structures. Sequential control structure. Alternative control structure. Alternative nested control structures.	Presentation with PowerPoint support and practical exemplification	
7	Previously conditioned repetitive control structure	Presentation with PowerPoint support and practical exemplification	
8	Repetitive control structure with predefined number of iterations. Using data ranges and data collections	Presentation with PowerPoint support and practical exemplification	
9	Complex Data Structures in Python: Lists, Tuples, and Vectors. Operations on data structures.	Presentation with PowerPoint support and practical exemplification	
10	Concepts of object oriented programming. Classes and objects	Presentation with PowerPoint support and practical exemplification	
11	Built-in Python functions and user-defined functions. Local variables and global variables	Presentation with PowerPoint support and practical exemplification	
12	Testing and debugging Python programs	Presentation with PowerPoint support and practical exemplification	
13	SPM – Structural Pattern Matching	Presentation with PowerPoint support and practical exemplification	
14	Recap	Presentation with PowerPoint support and practical exemplification	

Bibliography

- Guido van Rossum and the Python development team, Python Tutorial, 2022, https://docs.python.org/3/download.html
- Python documentation, 2022, docs.python.org
- Dragos Vespan, Course and seminar support, 2022, online.ase.ro
- Christian Mayer, Coee Break, Python - 50 Workouts to Kickstart Your Rapid Code Understanding in Python, 2018, https://blog.finxter.com/python-cheat-sheet/
- Swaroop C H, A byte of Python, 2022, http://www.swaroopch.com/notes/Python

8.2. L/P(L/P)		Teaching/Work methods	Recommendations for students
1	Introductory seminar. Python IDE installation, configuration and presentation	Interactive presentation	
2	Examples of conversion algorithms between different numbering bases	Hands-on exercises through individual assignments	
3	Creating and running Python programs. Using print and input functions.	Hands-on exercises through individual assignments	
4	Using variables, operators and expressions in Python.	Hands-on exercises through individual assignments	
5	Using int and float data types. Examples of single and double precision representation	Hands-on exercises through individual assignments	
6	Negation of complex sentences. Using Logical Operators in Python.	Hands-on exercises through individual assignments	
7	The if conditional statement. Using nested if conditional statements.	Hands-on exercises through individual assignments	
8	The repetitive while statement. The repetitive instruction for. Iterative processing.	Hands-on exercises through individual assignments	
9	Manipulating lists and tuples in Python.	Hands-on exercises through individual assignments	
10	Defining classes and objects in Python and using them	Hands-on exercises through individual assignments	
11	Defining user-created functions and using them within Python applications	Hands-on exercises through individual assignments	
12	Creating and debugging Python programs	Hands-on exercises through individual assignments	
13	Project presentation	Individual project presentation. Creating a Python program to solve a given problem	
14	Project presentation	Individual project presentation. Creating a Python program to solve a given problem	

Bibliography

- Guido van Rossum and the Python development team, Python Tutorial, 2022, https://docs.python.org/3/download.html
- Python documentation, 2022, docs.python.org
- Dragos Vespan, Course and seminar support, 2022, online.ase.ro
- Christian Mayer, Coee Break, Python - 50 Workouts to Kickstart Your Rapid Code Understanding in Python, 2018, https://blog.finxter.com/python-cheat-sheet/
- Swaroop C H, A byte of Python, 2022, http://www.swaroopch.com/notes/Python

9. Corroboration of the contents of the discipline with the expectations of the representatives of the epistemic community, of the professional associations and representative employers in the field associated with the programme

The course contents respond to the current labor market requirements for digital skills, providing a foundation of knowledge and skills in computational thinking and computer problem solving through programming in the Python language

10. Assessment

Type of activity	Assessment criteria	Assessment methods	Percentage in the final grade
10.1. L/P(L/P)	Involvement in seminar classes and their preparation. Solving the given assignments	The frequency, the interaction during the laboratory hours and the solving of the assignmentsare recorded	20.00
10.2. L/P(L/P)	Creating the individual project, creating a Python program to solve a given problem. The individual presentation of the project is a mandatory condition for passing the exam.	The compliance of projects and The compliance of Python projects and programs with the established requirements is evaluated programs with the established requirements is evaluated	20.00
10.3. Final assessment	Scoring each correct answer and analyzing the interpretation of practical problems according to predefined scales	Written exam or computer written exam with multiple choice questions and practical problems to be solved	60.00

10.4. Modality of grading	Whole notes 1-10
10.5. Minimum standard of performance	Summing up at least 50% of seminar points and 50% of final evaluation score

Date of listing,
08/28/2025

Signature of the discipline leaders,

Date of approval in the department

Signature of the Department Director,