ISM UNIVERSITY OF MANAGEMENT AND ECONOMICS

APPROVED BY
THE STUDY COMMISSION
Minutes No. 02-12-2025-13
as of 26th August 2025

COMPUTER PROGRAMMING

Course code	*IT103*
Compulsory in the programmes	*Finance, Economics and Data Analytics*
Level of studies	*Undergraduate*
Number of credits	*6 ECTS (46 in-class hours + 4 hours of examination, 112 individual work hours)*
Course coordinator (title and name)	*Rolandas Rimkevičius*
Prerequisites	-
Language of instruction	*English*

THE AIM OF THE COURSE

This course is based on Python and SQL languages and has two aims: acknowledge students with the art of programming, and to introduce to data analytics and data science. Through the course, students learn how to solve real-life problems in the most efficient way using algorithms. The course introduces problem-solving, which is inspired by the social sciences. At the end of the course students are expected to create a final project.

MAPPING OF COURSE LEVEL LEARNING OUTCOMES (OBJECTIVES) WITH DEGREE LEVEL LEARNING OBJECTIVES (See Annex I), ASSESSMENT AND TEACHING METHODS

Course level learning outcomes (objectives)	Learning objectives for BSc in Business Management	Assessment methods	Teaching methods
CLO1. To be able to process information and operate at multiple levels of abstraction.	BLO4.1. Students will be able to communicate reasonably in different settings according to target audience tasks and situations.	Homeworks, midterms, final project,	Lectures, seminars, individual and group work
CLO2. To be able to decompose IT problems into parts and solve them efficiently.	BLO4.3. Students will be able to convey their ideas effectively in a written paper.	Homeworks, midterms, final project	Lectures, seminars, individual and group work
CLO3. To be able to demonstrate proficiency in a software development environment.	BLO3.2. Students will be able to make decisions using appropriate IT tools.	Homeworks, midterms, final project	Seminars, individual and group work
CLO4. To be able to assess the correctness, design, and style of code.	BLO3.2. Students will be able to make decisions using appropriate IT tools.	Homeworks, midterms, final project	Seminars, group work
CLO5. To be able to evaluate the project's complexity and estimate required resources.	BLO1.2. Students will be able to conduct a contextual analysis to identify a problem associated with their discipline, to generate managerial options and propose viable solutions.	Final project	Group work

ACADEMIC HONESTY AND INTEGRITY

The ISM University of Management and Economics Code of Ethics, including cheating and plagiarism are fully applicable and will be strictly enforced in the course. Academic dishonesty, and cheating can and will lead to a report to the ISM Committee of Ethics. With regard to remote learning, ISM remind students that they are expected to adhere and maintain the same academic honesty and integrity that they would in a classroom setting.

The course's philosophy on academic honesty is best stated as "be reasonable." The course recognizes that interactions with classmates and others can facilitate mastery of the course's material. However, there remains a line between enlisting the help of another and submitting the work of another. This policy characterizes both sides of that line.

The essence of all work that you submit to this course must be your own. Collaboration on problem sets is not permitted except to the extent that you may ask classmates and others for help so long as that help does not reduce to another doing your work for you. Generally speaking, when asking for help, you may show your code to others, but you may not view theirs, so long as you and they respect this policy's other constraints. Collaboration during midterm assessments is forbidden. Collaboration on the course's final project is permitted to the extent prescribed by its specification.

Reasonable	Not reasonable
Communicating with classmates about problem sets' problems in English (or some other spoken language), and properly citing those discussions.	Accessing a solution to some problem prior to (re-) submitting your own.
Discussing the course's material with others in order to understand it better.	Accessing or attempting to access, without permission, an account not your own.
Helping a classmate identify a bug in their code at lectures, elsewhere, or even online, as by viewing, compiling, or running their code after you have submitted that portion of the problem set yourself. Add a citation to your own code of the help you provided and resubmit.	Asking a classmate to see their solution to a problem set's problem before (re-)submitting your own.
Incorporating a few lines of code that you find online or elsewhere into your own code, provided that those lines are not themselves solutions to assigned problems and that you cite the lines' origins.	Discovering but failing to disclose to the course's head bugs in the course's software that affect scores.
Reviewing past semesters' tests and final exams and solutions thereto.	Decompiling, deobfuscating, or disassembling the staff's solutions to problem sets.
Sending or showing code that you've written to someone, possibly a classmate, so that he or she might help you identify and fix a bug, provided you properly cite the help.	Failing to cite (as with comments) the origins of code or techniques that you discover outside of the course's own lessons and integrate into your own work, even while respecting this policy's other constraints.
Turning to the course's heads for help or receiving help from the course's heads during a final exam or test.	Giving or showing to a classmate a solution to a problem set's problem when it is he or she, and not you, who is struggling to solve it.
Turning to the web or elsewhere for instruction beyond the course's own, for references, and for solutions to technical difficulties, but not for outright solutions to problem set's problems or your own final project.	Looking at another individual's work during the final exam or test.
Whiteboarding solutions to problem sets with others using diagrams or pseudocode but not actual code.	Manipulating or attempting to manipulate scores artificially, as by exploiting bugs or formulas in the course's software.
	Paying or offering to pay an individual for work that you may submit as (part of) your own.
	Providing or making available solutions to problem sets to individuals who might take this course in the future.
	Searching for or soliciting outright solutions to problem sets online or elsewhere.
	Splitting a problem set's workload with another individual and combining your work.
	Submitting (after possibly modifying) the work of another individual beyond the few lines allowed herein.
	Submitting the same or similar work to this course that you have submitted or will submit to another.
	Submitting work to this course that you intend to use outside of the course (e.g., for a job) without prior approval from the course's heads.
	Turning to humans (besides the course's heads) for help or receiving help from humans (besides the course's heads) during the final exam or midterm.
	Viewing another's solution to a problem set's problem and basing your own solution on it.

COURSE OUTLINE

Topic	In-class hours	Readings
1. Python: introduction, variables. General information about programming languages, number systems, code conversion, algorithms. Primitive data structures: integers, floats, strings, booleans. *Homework 1.*	4	Lecture notes https://www.w3schools.com/python/default.asp (intro – operators)
2. Python: collections. Non-primitive data structures and their properties: tuples, lists, dictionaries, sets. *Homework 2.*	4	Lecture notes https://www.w3schools.com/python/default.asp (lists – dictionaries)
3. Python: loops. "If…", "while…", "for…" loops. *Homework 3.*	4	Lecture notes https://www.w3schools.com/python/default.asp (if, while, for)
4. Python: functions. Built-in functions, defining and calling your own functions, lambda functions, error handling. *Homework 4.*	4	Lecture notes https://www.w3schools.com/python/default.asp (functions, lambda)
5. Python: modules. Overview of modules, built-in modules, "import…". *Homework 5.*	4	Lecture notes https://www.w3schools.com/python/python_modules.asp
MIDTERM ASSESSMENT 1	**1**	
6. Python: Numpy module. Overview, arrays, methods. *Homework 6.*	4	Lecture notes https://www.w3schools.com/python/default.asp (arrays) https://www.w3schools.com/python/numpy/default.asp
7. Python: Pandas module. Overview, data reading, filtering, grouping and other functionalities. *Homework 7.*	4	Lecture notes https://www.w3schools.com/python/pandas/default.asp
8. Graphs with Python. Graphing with Pandas module, Matplotlib, Seaborn, other graphing techniques. *Homework 8.*	4	Lecture notes https://www.w3schools.com/python/pandas/pandas_plotting.asp https://www.w3schools.com/python/matplotlib_intro.asp https://seaborn.pydata.org/
9. SQL (1). Introduction to databases: structure of a database, extraction and filtration of data.	4	Lecture notes https://www.w3schools.com/sql/default.asp
10. SQL (2). Work with databases: complex queries (joins), creating, other functionalities. *Homework 9.*	6	
11. SQL + Python. Work with databases using Pyhon language. *Homework 10.*	4	Lecture notes https://www.w3schools.com/python/python_mysql_getstarted.asp
MIDTERM ASSESSMENT 2	**3**	
Total:	**50**	
Defense of the final project	**2 (for the entire class)**	

FINAL GRADE COMPOSITION

Type of assignment	**%**
Individual Components 50%	
Homework 1 (Python: variables)	1
Homework 2 (Python: collections)	1
Homework 3 (Python: loops)	1
Homework 4 (Python: functions)	1
Homework 5 (Python: modules)	1
Homework 6 (Python: Numpy)	1
Homework 7 (Python: Pandas)	1
Homework 8 (Python: graphs)	1
Homework 9 (SQL)	1
Homework 10 (Python + SQL)	1
Midterm assessment 1 (topics 1 – 5)	15
Midterm assessment 2 (topics 6 – 11)	25
Group Components 50%	
Final project	50
Total:	**100**

DESCRIPTION AND GRADING CRITERIA OF EACH ASSIGNMENT
(Provide short descriptions and grading criteria of each assignment)

The overall assessment of the course (total maximum of 100% is possible) will be composed from evaluations of multiple tasks (homeworks, two midterm assessments, final project), which are described as follows:

1. **Homeworks** are short skill-forming tasks, which students have to complete individually until deadlines. Each homework values 1% of the final grade, and there are 10 homeworks in total. Detailed rules for homeworks (how to complete, where to upload, evaluation criteria, etc.) and deadlines will be announced on elearning.
2. **Midterm assessments** are in-class open-book individual examinations in a computer classroom. Students have to solve real-life problems within the given time. Duration of the first midterm assessment is one academic hour. It covers topics 1-5 (Python syntax) and values 15% of the final grade. Duration of the second midterm assessment is three academic hours. It covers topics 6-11 (data analytics) and values 25% of the final grade. Detailed rules for each midterm (how to complete, where to upload, evaluation criteria, etc.) will be announced on elearning.
3. **Final project** is a group work, which will require a group of students to solve and to present (defend) a real-life problem assigned by a professor or a company/state institution. Each student must create and present their own code that contributed to the project. Evaluation will be individual, taking into account both the student's individual input and the overall group performance. Final project values 50% of the final grade. Detailed rules for the final project (division into groups, tasks, exact requirements, defense rules, evaluation criteria, etc.) will be announced on elearning.

RETAKE POLICY
(Provide short description and percentage of the final grade)

In case of the negative final evaluation, retake is possible. It will cover material of the whole course and will comprise **50%** of the final grade. Marks earned for midterm assessments and homeworks will be annulled. Retake is four academic hours long open-book individual examination in a computer classroom.

ADDITIONAL REMARKS

1. Precision of composite evaluations is left intact (up to 2 decimal places) until the end of semester and only the final evaluation will be subject to rounding.

2. If a student doesn't participate in the final project, his/her maximum evaluation for the course can be only 5 out of 10.

REQUIRED READINGS

1. Lecture notes.
2. https://www.w3schools.com

ADDITIONAL READINGS

1. Effective Python: 59 specific ways to write better Python, Brett Slatkin, 2015, ISBN 978-0-13-403428-7
2. Learning Python, Fifth edition, Mark Lutz, 2013, ISBN: 978-1-449-35573-9
3. Python Cookbook, Third Edition, David Beazley and Brian K. Jones, 2013, ISBN: 978-1-449-34037-7
4. Python Crash Course, Eric Matthes, 2016, ISBN-10: 1-59327-603-6
5. Python for Data Analysis, Second edition, Wes McKinney, 2017, ISBN: 978-1-491-95766-0
6. Think Stats: Exploratory Data Analysis in Python, Allen B. Downey, 2014,
7. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron, 2019, ISBN: 2019 978-1-492-03264-9;

MORE EXERCISES

1. https://www.hackerrank.com/domains/python
2. https://edabit.com/challenges
3. https://holypython.com/beginner-python-exercises/
4. https://www.hackerrank.com/domains/python
5. https://codesignal.com/
6. https://pythontutor.com/index.html

DATA FOR THE FINAL WORK

1. https://www.kaggle.com/datasets
2. https://github.com/awesomedata/awesome-public-datasets
3. https://data.gov.lt/
4. https://data.europa.eu/euodp/en/data/
5. https://atvira.sodra.lt/imones/rinkiniai/index.html

APPROVED BY
THE STUDY COMMISSION
Minutes No. 02-12-2025-13
as of 26th August 2025

ANNEX I

DEGREE LEVEL LEARNING OBJECTIVES

Learning objectives for the Bachelor of Business Management
Programmes:
International Business and Communication, Business Management and Marketing, Finance, Industrial Technology Management, Entrepreneurship and Innovation

Learning Goals	Learning Objectives
Students will be critical thinkers	BLO1.1. Students will be able to understand core concepts and methods in the business disciplines
	BLO1.2. Students will be able to conduct a contextual analysis to identify a problem associated with their discipline, to generate managerial options and propose viable solutions
Students will be socially responsible in their related discipline	BLO2.1. Students will be knowledgeable about ethics and social responsibility
Students will be technology agile	BLO3.1. Students will demonstrate proficiency in common business software packages
	BLO3.2. Students will be able to make decisions using appropriate IT tools
Students will be effective communicators	BLO4.1. Students will be able to communicate reasonably in different settings according to target audience tasks and situations
	BLO4.2. Students will be able to convey their ideas effectively through an oral presentation
	BLO4.3. Students will be able to convey their ideas effectively in a written paper