COURSE UNIT (MODULE) DESCRIPTION

Course unit (module) title
Numerical Methods I

Academic staff	Core academic unit(s)
Coordinating: Marijus Ambrozas **Other: Dr. Stepas Toliautas, Edvinas Gvozdiovas**	Faculty of Physics

Study cycle	Type of the course unit
First (bachelor studies)	Compulsory

Mode of delivery	Semester or period when it is delivered	Language of instruction
Face-to-face / distant learning	2nd semester	English

Requisites	
Prerequisites: **Computer Literacy for Physicists**	**Co-requisites (if relevant):** **None**

Number of ECTS credits allocated	Student’s workload (total)	Contact hours	Individual work
5	136	64	72

Purpose of the course unit
To develop students’ working and learning skills, as well as improve their programming skills, and build a basic understanding of numerical methods and their use cases.

Learning outcomes of the course unit	Teaching and learning methods	Assessment methods
Be able to use *Python* programming language for solving basic tasks	Interactive lectures, live coding demonstrations, coding exercise sessions, individual homework tasks	Homework, mid-term exam, final exam, project work.
Be able to plan and solve basic problems in computer science, mathematics, and physics using simple numerical methods.	Interactive lectures, live coding demonstrations, coding exercise sessions, individual homework tasks, individual or group project work	Homework, mid-term exam, final exam, project work.
Be able to search for, analyze and critically assess the available information on the topics of interest.	Coding exercise sessions, individual homework tasks, individual or group project work	Homework, mid-term exam, final exam, project work
Be able to learn and work individually and in groups.	Coding exercises, individual homework tasks, individual or group project work	Homework, mid-term exam, final exam, project work
Be able to describe and present their work, critically assess the work results, and make conclusions.	Coding exercises, individual or group project work, project work presentation	Project work

Content	**Contact hours**							**Individual work: time and assignments**	
	Lectures	Tutorials	Seminars	Workshops	Laboratory work	Internship	**Contact hours, total**	**Individual work**	**Tasks for individual work**
1. General course information and basic introduction to numerical modelling.	2				2		**4**	**1**	Setting up the necessary software
2. Refresher for the beginner topics of *Python* programming language: variables, lists, basic operations, flow control, loops, files, basic debugging	4				4		**8**	**4**	Literature studies, homework exercises
3. Intermediate topics of *Python* programming language: functions, recursion, classes, methods, operators	4				4		**8**	**6**	Literature studies, homework exercises
4. Refresher and intermediate topics of the most popular *Python* packages: *random*, *numpy*, *pandas*, *matplotlib*	2				2		**4**	**3**	Literature studies, homework exercises
5.Wrap-up of the programming basics, mid-term exam	2				2		**4**	**8**	Preparation for the mid-term exam
6. Root and extremum finding problems, and the most popular numerical algorithms to solve them	2				2		**4**	**3**	Literature studies, homework exercises
7. Linear algebra problems and the most popular numerical methods to solve them	3				3		**6**	**6**	Literature studies, homework exercises
8. Data interpolation, extrapolation, and fitting	2				2		**4**	**3**	Literature studies, homework exercises
9. Numerical derivatives and numerical integration	2				2		**4**	**4**	Literature studies, homework exercises
10. Statistics, Monte Carlo methods for physical system modelling	2				2		**4**	**4**	Literature studies, homework exercises
11. Numerical methods for differential equation solving	2				2		**4**	**2**	Literature studies, homework exercises
12. Practical examples of numerical modelling of physical systems	2				2		**4**	**2**	Literature studies, homework exercises
13. Wrap-up of the introduction to numerical methods, project work presentations	3				3		**6**	**26**	Literature studies, project work, preparing a presentation, preparation for the exam

Total	**32**				**32**		**64**	**72**	

Assessment strategy	Weight %	Deadline	Assessment criteria
Homework exercises	20	Throughout the semester	2 points – student understands and can apply all the main programming concepts for solving the problems using intermediate level numerical methods. 1 point – student understands and can apply the more basic programming concepts for solving problems using the simplest numerical methods.
Mid-term exam	30	By spring break	3 points – student understands and can apply all the main programming concepts while working independently. 2 points – student understands and can apply some intermediate level programming concepts but needs personal assistance for more complex tasks. 1 point – student understands and can apply only the most basic programming concepts without personal assistance.
Project work	20	By the final lecture of the semester	A student (alone or in pairs) must solve a basic physics task using a numerical method of choice, and make a presentation explaining the problem, its solution, showing the results and making conclusions. 2 points. Evaluation is done according to the following criteria: - Complexity of the solved problem - Complexity of the applied numerical algorithm - Demonstrated understanding and clarity of the presentation - Depth of the analysis of the results and meaningfulness of the conclusions
Exam	30	During the exam session	3 points – student understands and can apply all the most common numerical methods to solve problems while working independently. 2 points – student understands and can apply some of the most common numerical methods to solve problems but needs personal assistance for more complex tasks. 1 point – student understands and can apply only the most basic numerical methods without personal assistance.

Author (-s)	Publishing year	Title	Issue of a periodical or volume of a publication	Publishing house or web link
Required reading				
A. Sweigart	2020	Automate the Boring Stuff with Python	2nd edition	San Francisco: No Starch Press. Open-source version available on author’s

				webpage: https://automatetheboringstuff.com/#toc
J. Kiusalaas	2013	Numerical Methods in Engineering with Python 3	3rd edition	Cambridge: Cambridge University Press. PDF version available with VU licence: https://www.cambridge.org/core/books/numerical-methods-in-engineering-with-python-3/95151C37C2F427F30DC90FA619FE79F9
M. H. Kalos & P. A. Whitlock	2008	Monte Carlo methods	2nd revised and enlarged edition	Wiley-Blackwell. PDF version available with VU licence: https://onlinelibrary.wiley.com/doi/book/10.1002/9783527626212
Recommended reading				
E. Matthes	2016	Python Crash Course	3rd edition	No Starch Press.
A. Gezerlis	2020	Numerical Methods in Physics with Python.		Cambridge: Cambridge University Press. PDF version available with VU licence: https://www.cambridge.org/core/books/numerical-methods-in-physics-with-python/563DF013576DCC535668A100B8F7D2F9