COMPUTATION III

SYLLABUS
2023-2024

INSTRUCTOR INFORMATION	JOÃO FONSECA jpfonseca@novaims.unl.pt joaopfonseca.github.io DAVIDE FARINATI dfarinati@novaims.unl.pt LIAH ROSENFELD lrosenfeld@novaims.unl.pt
SCHEDULE	TP1: • Monday 11:30h – 13:00h P1: • Thursday 14h – 15h30 • Friday 15h30 – 17h00 P2: • Wednesday 14h – 15h30 • Thursday 14h – 15h30
OFFICE HOURS	Wednesday 11:30h – 13:00h (schedule appointment by email) Room TBD
CONTACT	Students should use Moodle platform for any communication with the course instructors. Homework and project submissions should also be done through Moodle.
DESCRIPTION	This course is designed to provide students with a solid understanding of object-oriented programming (OOP), covering essential concepts such as class and method definition, library utilization, and more, using the Python programming language. It is assumed that the students attended the programming courses taught during the first year of the underlying bachelor degree at NOVA IMS.
OBJECTIVES	At the end of the course, students should be able to: • Model a problem using the Unified Modeling Language (UML); • Set up and use an appropriate software development environment according to a project's needs; • Write and debug code based on OOP; • Write computer programs to solve specified problems; • Maintain, publish and document Python code;
COURSE SUCCESS	In this course, success depends on a number of factors: • A priori knowledge of Python programming; • Class attendance;

	• Work during the semester and not only when exams are about to start; • Develop the course project during the semester, making the most of the practical classes; • Complete exercises provided in class and consistent practice; • Read the suggested references and materials made available by the lecturer.
CONTENTS	1. Introduction to the course Computation III (Syllabus, Objectives, Project, Grading and Bibliography) 2. Re-introduction to programming with Python 3. Introduction to object oriented programming (OOP) 4. Introduction to unified modeling language (UML) and its application to software design 5. Inheritance and polymorphism in OOP 6. Introduction to versioning control system (VCS) with Git 7. Branching, undoing and forking in VCS 8. Examining and setting up Python libraries 9. Software development strategies 10. Creating computer games with PyGame
BIBLIOGRAPHY	References: ❑ Python Basics: A Practical Introduction to Python 3 (2020). Fletcher Heisler, David Amos, Dan Bader, Joanna Jablonski. ISBN: 9781775093329 (paperback), 9781775093336 (electronic). ❑ Object-oriented Python: Master OOP by Building Games and GUIs (2022). Irv Kalb. ISBN-13: 978-1-7185-0206-2 (print), 978-1-7185-0207-9 (ebook). **Note:** all references are available at NOVA IMS library or are provided by the teacher.
STUDENT EVALUATION	1st Call – 60% final exam, 40% practical group project 2nd Call – 100% final exam

Coursework

The project consists of the application of all contents discussed in the course. The students will develop a game using Python based on a set of specific guidelines. The guidelines provide the students with information on the type of tasks they should do and the general results they should achieve. The full description of the project guidelines, goals and requirements will be announced on lecture 6. The end product of the project should be your code, fully documented, and your project's UML diagram, published in an open-source repository. With this project, students are expected to develop their proficiency in Python programming, as well as understand software development best practices, documentation, maintenance and code distribution.

Project discussion: after submitting the projects, students will be called to discuss them with the instructors. Failing to attend the project discussion will result in a final grade of zero (to the individual student) in the group project.

Groups' composition: the project must be done as a group. The groups must have 3 students.

Project Deadline: December 20th **(Tentative)**

In-class behavior: Active and positive participation is an integral part of the learning experience and contributes significantly to both personal and group learning. As such, the teaching faculty reserves the right to apply penalties to the individual students' final grade for poor in-class behavior, at their discretion.

Tasks. In both practical and theoretical classes, students will be frequently assigned homework, which will consist of simple tasks related with the course material. It is expected that the students complete these tasks.

Final Exam (2nd call evaluation). The final exam will be a 1.5 hours in-class exam covering all the materials of the course.

Grading

1st Call	Final exam: 60% Project: 40%
2nd Call	Final exam: 100%

There are two opportunities to do the exam. Any delay in the delivery of the project is subject to a penalty of 10% of the grade for each day of delay. The project will be developed in groups, and each group must be composed of 3 elements. **To obtain approval in the course, the student cannot have less than 8 (40%) in the exam grade.**