

Introduction to Programming - Final Project

Projects are carried out individually.

There is no obligation to answer to all questions in the project description, and you should not be limited to such questions. The more you develop the problem in the project in elegant and interesting ways, the higher are the possibilities to receive a high mark.

1 Important steps:

- **Choose a project from the list below.** We are going to make the projects available on Moodle, and you will be able to associate your name to one of them. No more than three students can choose the same project.
- **What you need to submit:**
 - A **.py file** with your code in it. The code should be **clearly commented**.
 - A **short report** explaining the main aspects of your code (mathematical results, main functions, algorithms, complexity, issues, achievements, drawings, limits of your code...) as a **.pdf** file. Use the report to put your code into perspective.
The report should be written in L^AT_EX (there will be a full lecture on it), and it should be 2-3 pages long (indicatively). Insert python code the way explained during the lecture.
 - The **slides** for your presentation written in LaTeX (beamer) and submitted as a pdf file.
- **Deadline: 15 December at midnight.** You need to upload on Moodle all your files. Please name the files as:
 - Yourlastname-Youfirstname-slides.pdf ;
 - Yourlast Yourlastname-Youfirstname-reports.pdf ;
 - Yourlastname-Youfirstname-code.py

Presentation:

- You will present your project on **Monday, 16th of December**, during the last lecture.
- Each of you will have **8 minutes** available to present the project.
- In the presentation you need to introduce the problem, the ideas behind your solutions, how you have implemented it, the main problems you encountered and further work that needs to be done.

- **Clarity** and **concision** will be the most important aspects in evaluating your presentation.

2 List of Projects:

You pick up one project from the following list.

No more than three students can choose the same project.

Feel free to come to us with your own project, but make sure to have our approval before working on it.

The less you rely on existing modules, the more skills you show, and the higher the mark.

Project 1: Graphs

Mathematically, a graph is a finite set of vertices E along with a set of edges connecting pairs of elements of E . For the purpose of this project, start with non-oriented edges. You can begin with the definition of a class for graphs. After this you should compute the connected components of your graph and the smallest path between two points. You should also draw graphs. For more directions you might want to ask the same questions by looking at different kinds of graphs (weighted graphs, oriented graphs, probabilistic graphs).

Project 2: Integrals and ordinary differential equations (ODEs)

The aim of the project is to give various ways to compute the integral of a function. Drawings are welcome. Write functions to solve ordinary differential equations. Illustrate drawbacks and advantages of different methods.

Project 3: Simulation of probabilities

In this project, you may use the **random** module only to check that your code is correct. Find a way to simulate a random variable which, e.g., has the value 0 with probability 0.5 and 1 with probability 0.5. Use this random variable to simulate random variables with different laws (exponential, geometric, binomial, ...). Draw the empirical repartition functions.