

Guidelines for the Final Project

Imperative Programming with Python

January 2015

Proposal due date: 22/01/2015.

Final project due date: 01/02/2015.

The Proposal.

- You can choose the topic for the final project.
- By next the proposal due date your group (or a representative) has to send me (by mail) a final project proposal (pdf, txt). I'll read and propose changes in case I think it is not enough (or it is too much) or I think it is not appropriate for the course. The proposal should include: group members and mails, designated group representative (with whom I'll exchange mails), description of the problem to be solved.
- About the description: think that I'm an investor and I don't have your background (which is probably the case...the second part). Introduce all the necessary concepts for me to understand what the project is about and how to evaluate it.
- I uploaded some example project proposals that I accepted (after a few rewrites) in 2012. Check the webpage.
- For you to have an idea on the size of the final project, I expect the project to be (in estimated time/size) slightly more time-consuming than the homework sets.

The Project.

- The code has to be correctly separated in classes/functions/modules.
- I expect you to consider all the design concepts that we've seen (including those of HW3).
- I expect you to use the right data structures.
- I don't expect a program that "just works", that's the bare minimum. Your project should, if possible, take advantage of the Object Oriented Programming approach (this is possible 95% of the time).
- The code has to be properly commented.
- You should attach a report explaining, among others: the different parts of the system and what they do. Other possible design choices that you considered and why you didn't take them (if applicable), a user manual for the final program.
- You should include test cases, and explain how to evaluate the program.