

301AA - Advanced Programming [AP-24]

Master in Computer Science and Networking

[Master in Computer Science]

Department of Computer Science, Pisa

Academic Year 2024/25

Lecturer: **Prof. Andrea Corradini**

andrea.corradini@unipi.it

<http://pages.di.unipi.it/corradini/>



Course page: <http://pages.di.unipi.it/corradini/Didattica/AP-24/>

Virtual room on Teams: ***

to be used only in the event of a new pandemic

AP-01: Overview and Admins

Prerequisites

- Undergraduate level knowledge of
 - at least one object-oriented programming language (like Java, C++, C# or others)
 - at least one functional programming language (like OCaml, Scheme, Haskell or others)
- ➔ If you don't have this background, please inform me at the beginning of the course (in the form I'll ask you to fill out)
- ➔ Suggestions to fill possible gaps will be given

Goals of the course

- To provide the students with a deep understanding of how *high level programming concepts* and *metaphors* map into *executable systems* and which are *their costs and limitations*
- To gain familiarity with modern principles, techniques, and best practices of software construction
- To introduce the students to techniques of programming at higher abstraction levels, in particular *component programming* and *functional programming*
- To present *state-of-the-art frameworks* incorporating these techniques.

Organization of the course

- **Frontal lessons** in presence, using slides
- **Hands-on activities** will be organized to experiment with concepts, tools and languages presented in the lessons.
- Interaction with the lecturer: **during lessons**, by e-mails, in meetings during office hours (day/time to be fixed).
- On the **web page of the course**, the slides presented in each lesson are published progressively, with references to corresponding topics in the reading material.
- → <http://pages.di.unipi.it/corradini/Didattica/AP-24/>

Evaluation

Evaluation

- Some programming assignments during the course
 - *Suggested deadline* will be published together with the text of the assignments
 - **Real deadline** is 10 working days before the oral exam
- Final oral exam (in presence): by appointment, preferably during exam sessions

Attendance to the course is strongly encouraged

- If you miss a lesson, you can find on the course web page the list of topics presented, with slides and references to reading material
- You might even find a recording of the lesson: see next slide

Non-attending students

Exams for **non-attending students** are identical to those for attending students (in presence, programming ass. + oral)

Students who cannot attend the course for **valid reasons** (e.g. **visa problems**) have two choices:

- Stick to the 2024/25 academic year syllabus.
- Bring to the exam the 2021/22 academic year syllabus.
In addition to the teaching material, they can also find the recordings of the lessons held in that year on page <https://pages.di.unipi.it/corradini/Didattica/AP-21/>.

If a student wishes to bring the 2021/22 academic year syllabus, he or she must ask the teacher for authorization as soon as possible, and in any case not after the end of the course.

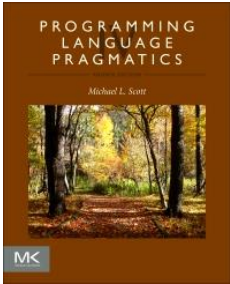
Reading material

- Will be suggested progressively along the course
- Mostly material accessible on-line

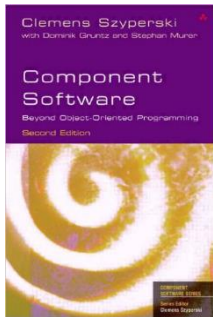
Credits

- Slides of the course freely taken and elaborated from a number of sources, including previous instances of this course (by [Giuseppe Attardi](#), [Gianluigi Ferrari](#), [Antonio Cisternino](#)) and others that will be indicated along the course

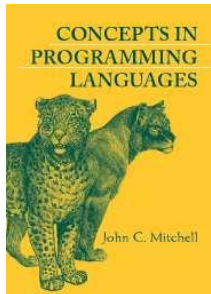
Some Suggested Readings



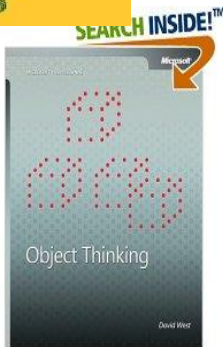
Programming Language Pragmatics, 4th ed. Michael L. Scott, Morgan-Kaufmann, 2015.



Component Software: Beyond Object-Oriented Programming. C. Szyperski, D. Gruntz, S. Murer, Addison-Wesley, 2002.



Concepts in Programming Languages. John C. Mitchell, Cambridge University Press, 2002.



Object Thinking. D. West, Microsoft Press, 2004.

<https://doc.rust-lang.org/book/>