301AA - Advanced Programming [AP-22]

Lecturer: **Andrea Corradini** andrea@di.unipi.it
http://pages.di.unipi.it/corradini/

Assistant: **Laura Bussi** laurabussi@live.it

Course pages: http://pages.di.unipi.it/corradini/Didattica/AP-22/

Virtual room: on Teams

Department of Computer Science, Pisa
Academic Year 2022/23

***AP-01**: Overview and Admins*

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.

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 Haskell, OCaml, Scheme or others)

➔ If you don’t have this background, please inform me at the beginning of the course

➔ Suggestions to fill possible gaps will be given

Programme

- Programming Language Pragmatics
- Run Time Support and Execution Environments: the Java Virtual Machine
- Components based programming and Frameworks
- Polymorphism: a classification and examples in several languages
- Functional languages: Haskell and advanced concepts
- Stream API and lambda-expressions in Java
- Ownership and Borrowing in Rust
- Scripting Languages and Python

Organization of the course

- **Frontal lessons** in presence (room L1 or A1), using slides
- **Hands-on activities** will be organized (with the help of Laura Bussi), 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-22/
- → see also http://pages.di.unipi.it/corradini/Didattica/AP-21/
- Recordings of the first lesson and of last year’s lessons will be made available

Evaluation and other things…

Evaluation

- Some programming assignments during the course
- Final oral exam

Attendance to the course is strongly encouraged

- *The recorded lessons are available for exceptional situations*
- 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

Examination methods for **non-attending students** are identical to those for attending students

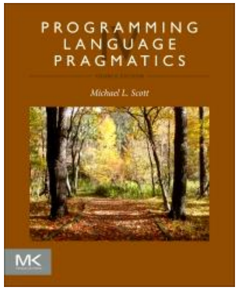
Reading material

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

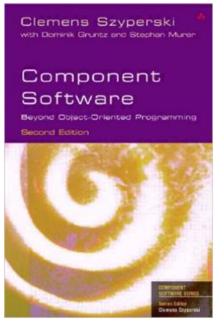
Credits

- 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

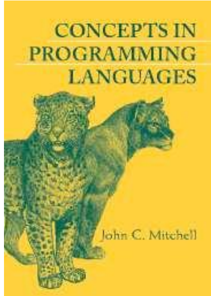
Some Suggested Readings



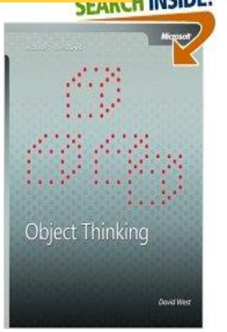
Programming Language Pragmatics, 4th ed. Michael L. Scott, Morgan-Kaufmann, 2015.


Clemens Szyperski
with Dominik Gruntz and Stephan Murer
Component Software
Beyond Object-Oriented Programming
Second Edition

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.