Teacher

Davide Ancona

office n. **328**, e-mail **davide.ancona@unige.it**

Objectives

- Learning the object-oriented and functional programming methodology, both at the conceptual and practical level
- Getting acquainted with object-oriented design and programming patterns
- Acquiring some basic concepts for designing and implementing a programming language

Motivations

Importance of software productivity, reuse and maintenance

Motivations

- What is the best programming language?
  - The question hardly makes sense in general
  - many considerations play an important role in the choice of the language
  - application domain, project requirements, developers' skills, software interoperability, compatibility issues, market requests, etc.
- Importance of mastering different computational models, several languages are in fact multi-paradigm

Motivations

- Compilers and interpreters are very basic tools: every computer professional and graduate in computer science/engineering should have a rough idea of how they are designed and implemented
- Domain specific languages (DSL) enhance software productivity, but require skills in programming language implementation to be developed

Motivations

Prepare your future career!

- Developer jobs: From SQL to Java, these are the skills companies are looking for now
  - `www.zdnet.com/article/developer-jobs-from-sql-to-java-these-are-the-skills-companies-are-looking-for`
- The Top Programming Languages 2019
  - `www.tiobe.com/tiobe-index`
  - `spectrum.ieee.org/computing/software/the-top-programming-languages-2019`

Exams

- Written test (see solutions of past exams)
- Optional oral test (questions from the syllabus), compulsory when written test mark between 15 and 17 (inclusive), or to obtain 30L
- Java project (2-3 students per project) + discussion
- Partial written test (compitino) during the Winter term (typically February)
- Written test: 5 exams, but no more than 3 tries
- All rules available on AulaWeb
- Requirements (prerequisiti): IP and ASD
- **Remark:** it is **strongly recommended** (but not compulsory) to pass IP and ASD exams first

Aulaweb

Linguaggi e Programmazione Orientata agli Oggetti (a.a. 2017/18)

Dibris / Corsi / Informatica (Scuola di Scienze MFN) / Laurea in Informatica / Anno Accademico 2017/18 / Secondo Anno / LPO-1718

- Announcements
- Forum per discussioni su questioni tecniche od organizzative
- Testi consigliati
- Modalità d'esame
- Link utili su Java
- Esami e soluzioni
- Questionario informativo sugli studenti frequentanti

25 settembre - 1 ottobre

2 ottobre - 8 ottobre

9 ottobre - 15 ottobre

16 ottobre - 22 ottobre

23 ottobre - 29 ottobre

30 ottobre - 5 novembre

6 novembre - 12 novembre

13 novembre - 19 novembre

20 novembre - 26 novembre

27 novembre - 3 dicembre

4 dicembre - 10 dicembre

Schedule (semester 1)

	Monday	Tuesday	Wednesday	Thursday	Friday
9-10	class/lab	class		class/lab	
10-11	class/lab	class		class/lab	
11-12					
12-13					
13-14					
14-15					
15-16					

Classes start at **9.20 am** and end at **10.55 am** with a quick break in the middle
Second semester: 2 hours of class/lab per week

Lab

- 2 OCaml labs, 8 Java labs
- applications: JDK (Java Development Kit) and Eclipse/IntelliJ IDEA, OCaml interpreter and OCaml-Top
- more advanced Java labs in the second semester

Lab

Advices on the Java project

There are just three very simple rules

Lab

Advices on the Java project

There are just three very simple rules

1. **attend the Java labs and complete the assignments at home**

Lab

Advices on the Java project

There are just three very simple rules

1. **attend the Java labs and complete the assignments at home**
2. **attend the Java labs and complete the assignments at home**

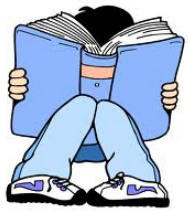
Lab

Advices on the Java project

There are just three very simple rules

1. **attend the Java labs and complete the assignments at home**
2. **attend the Java labs and complete the assignments at home**
3. **attend the Java labs and complete the assignments at home**

Suggested readings



Basic

- D. Ancona, G. Lagorio, and E. Zucca. Linguaggi di Programmazione. Città Studi Edizioni.
- K. Arnold, J. Gosling, and D. Holmes. The Java Programming Language, Fourth Edition, Prentice Hall.
- E. Chailloux, P. Manoury, and B. Pagano. Developing Applications With Objective Caml. (freely available on the Web)

Advanced

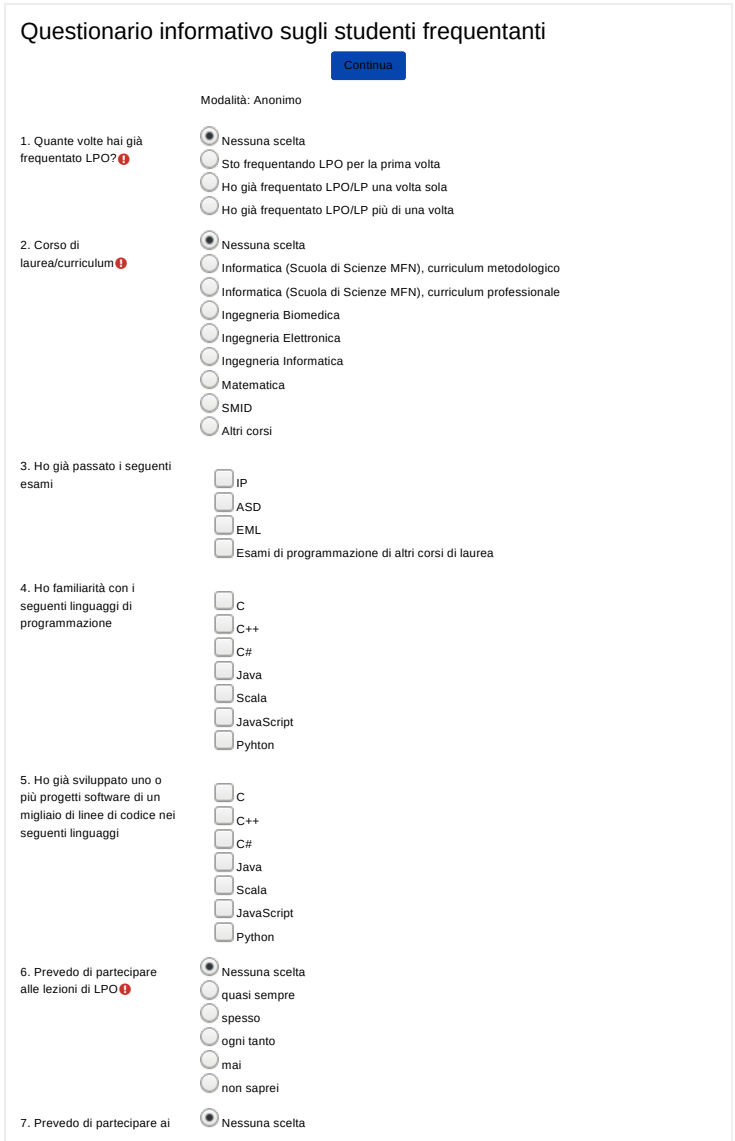
- J. Bloch. Effective Java (2nd Edition). Prentice Hall.
- E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley.

Technical (useful for consultation only)

- J. Gosling, B. Joy, G. Steele, G. Bracha, A. Buckley. The Java Language Specification (Java SE, latest edition). Oracle. (freely available on the Web)

Students introduce themselves

Please submit the short AulaWeb feedback



Questionario informativo sugli studenti frequentanti

Continua

Modalità: Anonimo

1. Quante volte hai già frequentato LPO?
- (•) Nessuna scelta
- ( ) Sto frequentando LPO per la prima volta
- ( ) Ho già frequentato LPO/LP una volta sola
- ( ) Ho già frequentato LPO/LP più di una volta

2. Corso di laurea/curriculum
- (•) Nessuna scelta
- ( ) Informatica (Scuola di Scienze MFN), curriculum metodologico
- ( ) Informatica (Scuola di Scienze MFN), curriculum professionale
- ( ) Ingegneria Biomedica
- ( ) Ingegneria Elettronica
- ( ) Ingegneria Informatica
- ( ) Matematica
- ( ) SMID
- ( ) Altri corsi

3. Ho già passato i seguenti esami
- [ ] IP
- [ ] ASD
- [ ] EML
- [ ] Esami di programmazione di altri corsi di laurea

4. Ho familiarità con i seguenti linguaggi di programmazione
- [ ] C
- [ ] C++
- [ ] C#
- [ ] Java
- [ ] Scala
- [ ] JavaScript
- [ ] Pyhton

5. Ho già sviluppato uno o più progetti software di un migliaio di linee di codice nei seguenti linguaggi
- [ ] C
- [ ] C++
- [ ] C#
- [ ] Java
- [ ] Scala
- [ ] JavaScript
- [ ] Python

6. Prevedo di partecipare alle lezioni di LPO
- (•) Nessuna scelta
- ( ) quasi sempre
- ( ) spesso
- ( ) ogni tanto
- ( ) mai
- ( ) non saprei

7. Prevedo di partecipare ai
- (•) Nessuna scelta

Syllabus

A general outline

- Programming language principles (part 1)
  - Syntax: regular expressions, and context-free grammars
- Functional programming in OCaml
- Object-oriented programming and Java
- Programming language principles (part 2)
  - Parser implementation
  - Static and dynamic semantics and their implementation

A more detailed description can be found on AulaWeb

Principles of Programming Languages

Two important problems

- how to provide a precise definition of a programming language?
- how to implement a higher-level programming language?

Formal Specification of Programming Languages

Main parts of a programming language specification

- syntax
- (optional) static semantics
- dynamic semantics