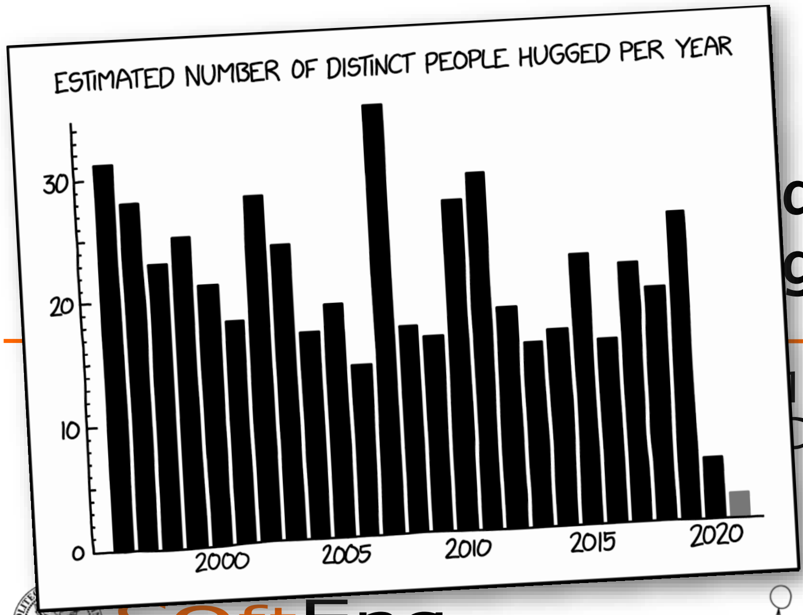
Object-Oriented Programming

2020-21
04JEYLM, 04JEYOA, 04JEYPC

SoftEng
http://softeng.polito.it



Staff

- Giovanni Squillero
  `giovanni.squillero@polito.it`
- Giuseppe Rizzo
  `giuseppe.rizzo@linksfoundation.com`

Staff

- Giovanni Squillero

  `giovanni.squillero@polito.it`

- Giuseppe Rizzo

  `giuseppe.rizzo@linksfoundation.com`

Topics

- Java programming language
  - Java syntax
  - Standard libraries
- Software Engineering
  - Software Life Cycle
  - Design
  - Test
  - Configuration management
  - Object-oriented paradigm

Objectives

- Learn the Java language
- Write and test simple Java programs
- Use the development support tools
- Understand how software development works
- Become familiar with the basic development support instruments

Objectives

- Learn the Java language
- Write and t imple Java programs
- ort tools
-
- Become familiar with the basic development support instruments

Language shapes the way we think, and determines what we can think about.
– Benjamin Lee Whorf

Organization of the course

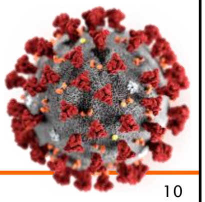
- Lectures (~50h)
  - Java (~35h)
  - Software Engineering (~15h)
- Classroom exercises (~20h)
  - Examples (~10h)
  - Assignments solutions (~10h)
- Lab assignments (~20h)
  - Three hours slots

Labs

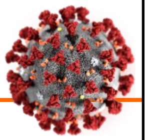
- LAIBs
  - 1.5h with Instructor / Student Assistants
  - 1.5h with Student Assistants

CANCELLED

- Assignments
  - Programs to be completed/modified
  - Similar process as in the final exam
- Assessed but not graded
- Essential for final exam
  - the only way to learn programming is programming

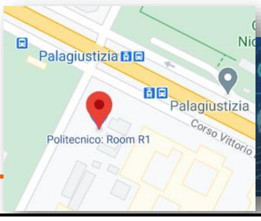


Schedule



- Lecture:
  - Monday @ 08:30-11:30 Cyberspace
  - Wednesday @ 16:00-17:30 Room R1
  - Friday @ 08:30-10:00 Cyberspace
- Lab:
  - Friday @ 11:30-14:30 Cyberspace

week 3?



Telegram

- Informal chat:
  `https://t.me/joinchat/WVdHwhFZ7pZO8CYs`

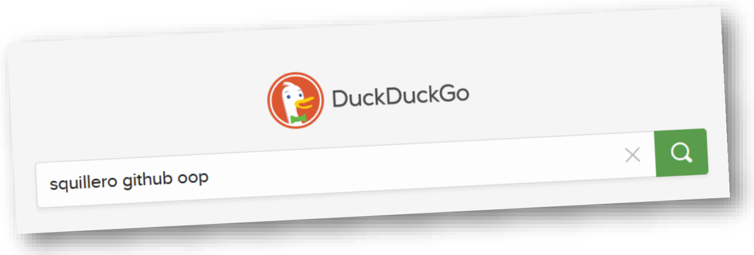
SoftEng http://softeng.polito.it

Material

- All materials is available from `http://www.polito.it/`

Material (cont'ed)

- All code samples will be available
  - oop.polito.it (subversion)
  - GitHub (git)

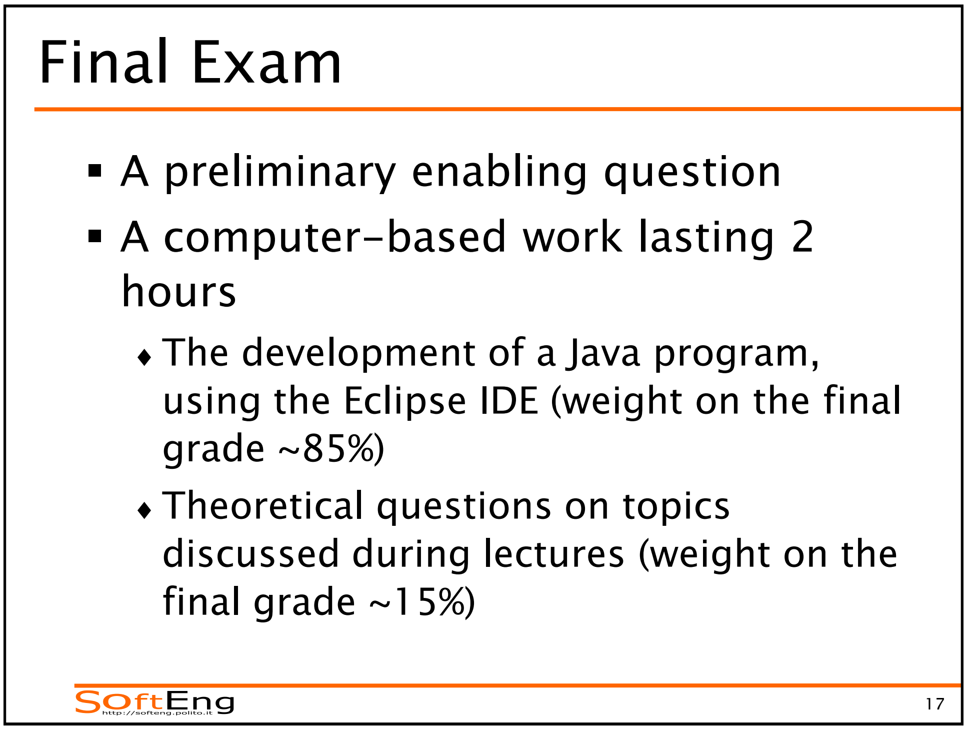


Requirements

- Mandatory
  - Procedural programming (that is: C)
- Recommended
  - Abstract data types
    - Lists, trees etc.
  - Algorithms
    - Sort, search, list insert etc.

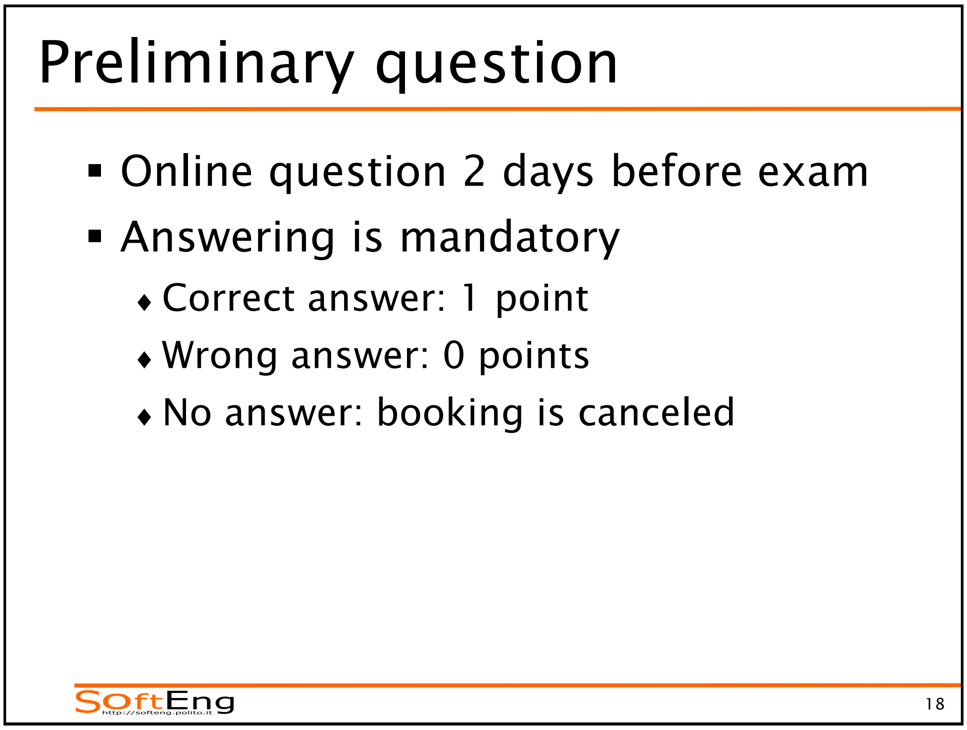
Software

- Mandatory
  - Java 11
    `https://docs.aws.amazon.com/corretto/latest/corretto-11-ug/`
  - Eclipse IDE (Java IDE)
    `http://www.eclipse.org/ide/`
  - Subversive plug-in for Eclipse
    Through Eclipse marketplace



Final Exam

- A preliminary enabling question
- A computer-based work lasting 2 hours
  - The development of a Java program, using the Eclipse IDE (weight on the final grade ~85%)
  - Theoretical questions on topics discussed during lectures (weight on the final grade ~15%)



Preliminary question

- Online question 2 days before exam
- Answering is mandatory
  - Correct answer: 1 point
  - Wrong answer: 0 points
  - No answer: booking is canceled

Theory Questions

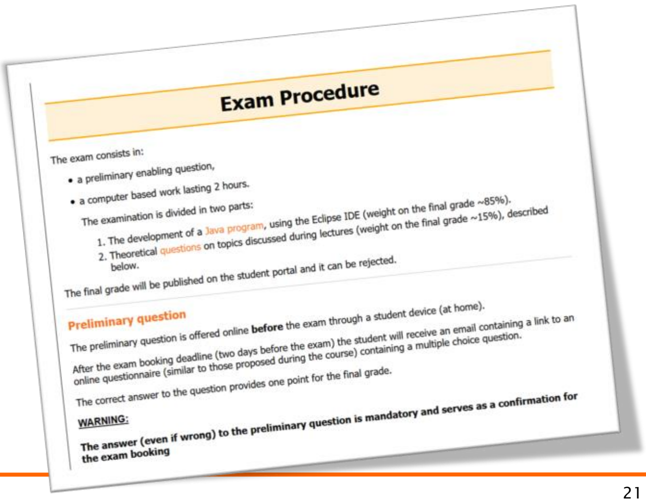
- Score
  - no answer: 0 points
  - perfectly correct answer: 1 point
  - completely wrong answer: -0.5 points

Programming part

- In the lab
  - Develop Java application, given
    - a textual specification of requirements
    - a skeleton code for the main functions
  - ♦ Submit initial version
- At home
  - ♦ Receive acceptance tests
  - ♦ Fix the app
  - ♦ Submit final version within a 3-7 days

Final Exam – Full Info

https://oop.polito.it/doc/Exam_en.html



Exam Procedure

The exam consists in:

- a preliminary enabling question,
- a computer based work lasting 2 hours.

The examination is divided in two parts:

1. The development of a Java program, using the Eclipse IDE (weight on the final grade ~85%).
2. Theoretical questions on topics discussed during lectures (weight on the final grade ~15%), described below.

The final grade will be published on the student portal and it can be rejected.

Preliminary question

The preliminary question is offered online **before** the exam through a student device (at home).

After the exam booking deadline (two days before the exam) the student will receive an email containing a link to an online questionnaire (similar to those proposed during the course) containing a multiple choice question.

The correct answer to the question provides one point for the final grade.

WARNING:

The answer (even if wrong) to the preliminary question is mandatory and serves as a confirmation for the exam booking

Readings – Java

- Java Documenation
  - http://www.oracle.com/technetwork/java/javase/documentation/index.html
- Arnold, Gosling, Holmes. **The Java Programming Language – 4th edition**, Addison-Wesley, 2006
- R. Urma, M. Fusco, A. Mycroft. **Java 8 in Action: Lambdas, streams, and functional-style programming.** Manning, 2015.
- Eckel. **Thinking in Java.** Prentice Hall, 4th Ed., 2006
  - www.mindview.com/Books

Readings – Sw Engineering

- Bruegge, Dutoit. **Object-Oriented Software Engineering Using UML, Patterns, and Java.** Pearson, 2009
- **ISO/IEC/IEEE Std 12207-2008 for Systems and Software Engineering – Software Life Cycle Processes**
  - http://ieeexplore.ieee.org/document/4475826/

Readings – Testing

- **ISO/IEC/IEEE, Std 29119-1 Software and systems engineering – Software testing – Part 1: Concepts and definitions, 2013.**
- **ISTQB, Certified Tester Foundation Level Syllabus, 2001**
  - http://www.istqb.org/downloads/send/2-foundation-level-documents/3-foundation-level-syllabus-2011.html4

Readings – Config Management

- Collins-Sussman, Fitzpatrick, Pilato. **Version Control with Subversion**, 2001
  - http://svnbook.red-bean.com
- **IEEE Std 828-2012 Standard for Configuration Management in Systems and Software Engineering, 2012**
- Semantic Versioning
  - http://semver.org

Readings – Design

- M.Fowler, K. Scott. **UML Distilled.** 3rd ed. Addison-Wesley, 2003.
- E. Gamma, R. Helm, R. Johnson, and J. Vlissides. **Design Patterns: Elements of Reusable Object-Oriented Software.** Reading, MA: Addison-Wesley, 1995.
- E.Freeman, E.Freeman, K.Sierra, B.Bates. **Head First Design Patterns.** O'Reilly, 2004