## COURSE TITLE
Programming Project

## COURSE CODE
76204

## SCIENTIFIC SECTOR
INF/01

## DEGREE
Bachelor in Computer Science

## SEMESTER
2nd

## YEAR
1st

## CREDITS
9

## TOTAL LECTURING HOURS
60

## TOTAL LAB HOURS
30

## ATTENDANCE
Attendance to course lectures and labs is optional. However, non-attending students have to contact the lecturer at the start of the course to discuss the modality of their independent study.

The exam modality for both attending and non-attending students is the same, which is described in the fields “Assessment” and “Evaluation criteria and criteria for awarding marks” below.

## PREREQUISITES
Students should be familiar with the basic knowledge of object-oriented programming and Java, as taught in the course “Computer Programming”

## COURSE PAGE
https://ole.unibz.it/

## SPECIFIC EDUCATIONAL OBJECTIVES
Type of course: “caratterizzanti” for L-31
Scientific area: “Discipline informatiche” for L-31

The course is designed to give specific professional skills. It will provide students with advanced techniques in Java. In particular, students will acquire knowledge in the overall architecture and components of Java SDK and JRE, use of advanced programming techniques (e.g. multi-threads, reading/writing streams, generics, regular expressions, exception handling and testing) and code documentation (e.g. generate API documentation) of the software developed.

## LECTURER
Tiago Prince Sales

## SCIENTIFIC SECTOR OF THE LECTURER
INF/01
# TEACHING LANGUAGE
- English

# OFFICE HOURS
- Tuesdays, from 08:00 to 10:00, by prior email appointment
  - tiago.princesales@unibz.it
  - Office POS 3.05 Faculty of Computer Science, Piazza Domenicani 3.

# TEACHING ASSISTANT
- Tiago Prince Sales
- Lanti Davide

# OFFICE HOURS
- Lanti Davide: Mondays 16:00-18:00 by prior email appointment, office POS 2.06, Faculty of Computer Science, Piazza Domenicani 3
davide.lanti@unibz.it

# LIST OF TOPICS COVERED
- Memory models in Java
- Virtual functions, late binding, overriding, and overloading
- Exception handling
- Reflection and runtime type identification
- Generics and collections
- I/O, serialization and XML/JSON processing
- Designing large applications: design patterns
- Multithreading
- Code optimization

# TEACHING FORMAT
- Frontal lectures, lab exercises, group projects.

# LEARNING OUTCOMES
- **Knowledge and understanding**
  - Know in details the fundamental principles of programming.
  - Have a solid knowledge of the most important data structures and programming techniques.

- **Applying knowledge and understanding**
  - Be able to develop small and medium size programs using different programming languages and paradigms.
  - Be able to solve problems through the application of programming methodologies.

- **Making judgments**
  - Be able to collect and interpret useful data and to judge information systems and their applicability.
  - Be able to work autonomously according to the own level of knowledge and understanding.

- **Communication skills**
  - Be able to use one of the three languages English, Italian and German, and be able to use technical terms and communication appropriately.
  - Be able to structure and write scientific documentation.

- **Learning skills**
  - Have acquired learning capabilities to pursue further studies with a high degree of autonomy.

# ASSESSMENT
- The assessment is based on:
  - weekly **lab exercises**, which are focused on specific topics taught in the course. They are meant to motivate students to study throughout the semester and consolidate the theoretical concepts taught in class;
• a **group project**, which evaluates if students acquired the expected programming knowledge and skills; and

• an **oral exam**, which evaluates if students assimilated of the theoretical concepts taught in class by reviewing and discussing the group project.

<table>
<thead>
<tr>
<th>ASSESSMENT LANGUAGE</th>
<th>English</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>EVALUATION CRITERIA AND CRITERIA FOR AWARDING MARKS</th>
<th>Final marks will be calculated in the following way:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• up to 10 points will be awarded to weekly assignments;</td>
</tr>
<tr>
<td></td>
<td>• up to 60 points will be awarded to the group project;</td>
</tr>
<tr>
<td></td>
<td>• up to 30 points will be awarded to the oral exam;</td>
</tr>
</tbody>
</table>

To enroll in the oral exam, a student must:

• Deliver all weekly assignments;

• Deliver the group project;

• Have earned 30 points from the weekly assignments and the group project combined.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Lecture notes handed out during the course</td>
</tr>
</tbody>
</table>

|-----------------------|--------------------------------------------------------------------------------------------------|


**SOFTWARE USED**

- JDK 13