

27024 - Computer Science II

Syllabus Information

Academic Year: 2020/21

Subject: 27024 - Computer Science II

Faculty / School: 100 - Facultad de Ciencias

Degree: 453 - Degree in Mathematics

ECTS: 6.0

Year: 4

Semester: First semester

Subject Type: Optional

Module: ---

1.General information

1.1.Aims of the course

1.2.Context and importance of this course in the degree

1.3.Recommendations to take this course

2.Learning goals

2.1.Competences

2.2.Learning goals

2.3.Importance of learning goals

3.Assessment (1st and 2nd call)

3.1.Assessment tasks (description of tasks, marking system and assessment criteria)

4.Methodology, learning tasks, syllabus and resources

4.1.Methodological overview

The methodology followed in this course is oriented towards the achievement of the learning objectives. A wide range of teaching and learning tasks are implemented, such as lectures, problem-solving sessions and autonomous work and study.

4.2.Learning tasks

This course is organized as follows:

- **Lectures.** Two weekly hours. The teacher will present the theoretical and technological concepts in lectures.
- **Problem-solving sessions.** Two weekly hours. Problems will be solved individually and by groups. These sessions involve the use of computers.
- **Autonomous work and study.**

The teaching activities and assessment tasks will take place in a face-to-face mode, except in the case that, due to the health situation, the dispositions emitted by the competent authorities and by the University of Zaragoza compel to take them in a telematic form.

4.3.Syllabus

This course will address the following topics:

- **Topic 1.** The object-oriented paradigm (OOP). The Java Virtual Machine: compiling and executing programs.
- **Topic 2.** Review of elements of structured programming in the Java language: predefined types; variables and constants; operators and expressions; usual mathematical functions (the Math class). Structured statements: sequential, conditional and iterative. Defining and invoking class methods. Method signatures: overloading methods.
- **Topic 3.** Introducing OOP. Objects, classes and references (the null reference). The life cycle of Java objects: the new operator and constructor methods, accessing members and message passing, the Java garbage collector. Array objects.
- **Topic 4.** Defining classes. Instance and class members. Writing constructor methods. Access levels: the public interface of a class. Namespaces: Java packages.
- **Topic 5.** An introduction to UML class diagrams. Class associations and associative classes. Roles and navigation.
- **Topic 6.** Inheritance: concept and types, method overriding. Class hierarchy: the Java Object class. Polymorphism: virtual methods. Modeling (generalization and specialization): abstract classes and methods.
- **Topic 7.** The Java type system: Java interfaces. Generic programming in Java. The Java collections framework.
- **Topic 8.** Exceptions: rising, handling and specification.
- **Topic 9.** Persistence: binary and text streams. Object persistence: the Serializable interface. Access to remote resources: File and URL classes.
- **Topic 10.** Event-driven programming: the Java event model. GUIs programming: containers, menus and basic controls.

4.4.Course planning and calendar

Further information concerning the timetable, classroom, office hours, assessment dates and other details regarding this course will be provided on the first day of class or please refer to the Faculty of Sciences website and Moodle.

4.5.Bibliography and recommended resources

http://biblos.unizar.es/br/br_citas.php?codigo=27024&year=2020