1. OVERVIEW

Subject area	Fundamentals of Programming
Degree	Bachelor's Degree in Computer Engineering
School/Faculty	Architecture, Engineering and Design
Year	2022 – 2023
ECTS	6 ECTS
Type	Compulsory
Language(s)	Spanish
Delivery mode	On campus / Online
Semester	7. ASSESSMENT
Year	7. ASSESSMENT
Coordinating professor	Ana del Valle Corrales Paredes

2. INTRODUCTION

This is the first of seven subject areas in the "Programming" subject. As this is the first subject area, no prior knowledge is expected and it aims to lay the foundations for students to learn about environments that involve programmable interfaces. Algorithmic thinking is introduced, so that it can be adapted to any programming environment and prepares for "Object Oriented Programming" (2nd programming subject). The fundamentals of programming will be introduced by using the Java programming language. The principals and good practices that will be acquired in the subject area are essential to lay a good foundation for expanding knowledge in the future. Algorithmics, as well as correct implementation and modularity, will enable the students to develop large programs through the use of abstraction.

Programming is not "something that is studied", but mastered through a lot of practice and referencing (programming language and environment manuals). For this reason, the content is highly focused on putting into practice the knowledge gathered in the documents and videos of the subject area. Through experimentation and the application of "good practices", students will reach a good level in the development of computer programmes and solutions, which will be practised through the activities proposed over the course of the subject area.

Good habits, as well as the ability to solve problems by creating code, will be tremendously useful and necessary in a wide range of subjects in the degree. Therefore, it can be concluded that Fundamentals of Programming is one of the fundamental pillars for the studies of the Computer Engineering degree.

3. SKILLS AND LEARNING OUTCOMES

Basic skills (CB, by the acronym in Spanish):

- CB1: Students have demonstrated knowledge and understanding of a study area originating from general secondary school education, and are usually at the level where, with the support of more advanced textbooks, they may also demonstrate awareness of the latest developments in their field of study.

- CB2: Students can apply their knowledge to their work or vocation in a professional manner and possess the skills which are usually evident through the forming and defending of opinions and resolving problems within their study area.
- CB3: Students have the ability to gather and interpret relevant data (usually within their study area) to form opinions which include reflecting on relevant social, scientific or ethical matters.
- CB4: Students can communicate information, ideas, problems and solutions to both specialist and non-specialist audiences.
- CB5: Students have developed the learning skills necessary to undertake further study in a much more independent manner.

Transversal skills (CT, by the acronym in Spanish):

- CT2: Self-confidence: Ability to evaluate one's own results, performance and skills with the self-determination necessary to complete tasks and meet any objectives.
- CT11: Planning and time management: Ability to set objectives and choose the right means to fulfil them through efficient use of time and resources.
- CT15: Responsibility: Ability to fulfil commitments to themselves and others when undertaking a task and try to meet a range of objectives within the learning process. Ability to face and accept the consequences of actions taken freely.
- CT16: Decision-making: Ability to choose between different options or methods to effectively solve varied situations or problems.
- CT17: Teamwork: Ability to integrate and collaborate actively with other people, departments and/or organisations in order to reach common goals.

Specific skills (CE, by the acronym in Spanish):

- CE3: Ability to understand and master the basic concepts of discrete mathematics, logic, algorithmics and computational complexity, and their application to solve engineering problems.
- CE5: Knowledge of the structure, organisation, operation and interconnection of computer systems, the fundamentals of their programming, and how they are used to solve engineering problems.
- CE12: Knowledge and application of the basic algorithmic procedures of computer technologies to solve problems, analysing the suitability and complexity of the proposed algorithms.
- CE13: Knowledge, design and efficient use of the most appropriate data types and structures to solve a problem.

Learning outcomes (RA, by the acronym in Spanish):

- RA1: Use simple (numeric, character, Boolean, etc.) and structured (arrays, records) data types.
- RA2: Apply the main components that make up a computer program: variables, constants, selections, repetitions, functions, etc.
- RA3: Implement algorithms in programs using a programming language.
- RA4: Manage data stored in files.
- RA5: Analyse different algorithmic solutions to a problem in order to choose the one that best meets specific needs, such as efficiency.
- RA6: Use data structures and algorithms to solve specific problems.

The following table shows how the skills developed in the subject area match up with the intended learning outcomes:

Skills	Learning outcomes
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17**	**7. ASSESSMENT**
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17, CE5, CE12**	**7. ASSESSMENT**
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17, CE3, CE5, CE12, CE13**	**7. ASSESSMENT**
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17, CE5, CE13**	**7. ASSESSMENT**
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17, CE3, CE5, CE12, CE13**	**7. ASSESSMENT**
CB1, CB2, CB3, CB4, CB5 **CT2, CT11, CT15, CT16, CT17 CE12, CE13**	**7. ASSESSMENT**

4. CONTENTS

The general contents of this subject area focus on the following topics:

- Elementary program design and mechanisms of abstraction.
- Basic data types and simple data structures.
- Basic constructs.
- Input / output

Introduction and history of the Object-Oriented paradigm.

- Core class design: Classes, relations, common structures.

The module is organised into 6 Learning Units:

Unit 1. Introduction to programming.

This unit introduces students to the Java programming language and how to install it with all its dependencies (execution and development platforms, JRE - JDK). This process will be validated through the implementation of our first program. “Hello World!”. Students will also work on basic aspects of programming: Basic data types, code comments, expressions and statements, differences between variables and constants, as well as value assignment.

Contents of the unit:

Introduction to Java

Data types

Variables, constants and assignment

Data input from console

Unit 2. Decisions

This unit introduces students to the control structures that enable different execution flows in programs. It also introduces relational and logical operators, as well as their behaviour depending on the evaluated data, since they will be necessary in control structures. In particular, students will work with if-then-else and switch-case control structures, as well as the Java conditional ? operator.

Contents of the unit:

Decisions with IF-THEN-ELSE

Decisions with SWITCH

Operator ?:

Unit 3. Loops

This unit introduces students to the need for iterative statement execution in programming. For this reason, the alternatives available for creating loops will be differentiated and the possible errors that can occur by configuring these loops will be explained in order to avoid them. In addition, students will work on understanding the close relation between data collections (arrays) and the loops for processing them.

Contents of the unit:

- For loops
- While and Do-While loops

Unit 4. Introduction to data structures: Arrays and Matrices

This unit will give an introduction to static data structures: Arrays and Matrices. Students will learn how to create, access and modify finite element structures.

Students will use knowledge from the previous unit to learn how to iterate over these.

Contents of the unit:

- Unidimensional arrays
- Multidimensional arrays or matrices.

Unit 6. Functions

This unit introduces the need to use and code functions for the correct modularization of programs. For this reason, student will review their components, as well as how to code

and use them in the Java programming language. This unit will also introduce the fundamentals and use of Object-Oriented Programming and basic class design.

• Contents of the unit:

• Parameterization of code

• Introduction to Object-Oriented Programming. Classes and objects.

Unit 5. Input / Output

This unit will introduce the need to input and output data to and from programs, known in computer science as the Input / Output process. For this reason, we will work with the alternatives available for data input (using files), as well as with the alternatives for data output in Java.

• Contents of the unit: File handling

5. TEACHING/LEARNING METHODS

The types of teaching/learning methods are as follows:

MD1. Survey on aims and interests. This survey is used to establish the aims of the subject and gather the student's interests on the subject. We will then make reference to it throughout the year for the students to evaluate the achievement of the aims and interests.

MD2. Lectures, subjects of study and seminars. The "lectures" taught in the on-campus delivery mode are called subjects of study and seminars in the online delivery mode, and are conducted through readings on the topic, technical notes and webinars (which are recorded for students to access).

MD3. Laboratory work: the laboratories will mainly be used in the on-campus delivery mode.

MD4. a) Group research and/or b) group problem-solving. This learning method will be used for the development of both declarative and procedural knowledge. In method type a), a different topic will be assigned to each group to be investigated. Later, new groups will be formed with students who have all studied a different topic, and these new groups will be proposed comprehension and problem-solving activities. In method type b), a series of short questions and problems will be proposed to be solved in groups.

MD5. Designs, understood as practical proposals for solving specific problems (unlike the study of practical cases, it is not a question of delving deeper into the analysis and the real problems. Instead, it is based on this knowledge, and the aim is to provide new solutions in accordance with engineering standards). These learning methods will be used for the development of procedural knowledge. They will be used in all delivery modes and help to develop creative potential and technical skills in the field of engineering.

MD8. Fieldwork, conferences, visits to companies and institutions. These will be used for the development of conditional knowledge. In the on-campus delivery mode, all learning methods may be used, while only conferences can be used in the online delivery mode, as they will be available for remote access in real time (via streaming technologies) or recorded and broadcast afterwards.

6. LEARNING ACTIVITIES

The types of learning activities, plus the amount of time spent on each activity, are as follows:

On campus:

Learning activity (AF, by the acronym in Spanish)	Number of hours
Lectures, reading on main topics and complementary materials, implementation of activities carried out independently and collectively.	50
Group work	25
Independent working	50
Tutorials, academic monitoring and assessment	25
TOTAL	**150**

Online:

Learning activity (AF, by the acronym in Spanish)	Number of hours
Independent working	50
Independent reading on complementary topics and materials and implementation of activities carried out independently. Asynchronous group discussion on the Campus Virtual forum, and online seminars	50
Group work	25
Tutorials, academic monitoring and assessment	25
TOTAL	**150**

7. ASSESSMENT

The assessment systems, plus their weighting in the final grade for the subject area, are as follows:

On campus:

Assessment system	Weighting
On-campus knowledge tests	30%
Reports and papers	15 - 30 %
Alternative assessment methods	15 - 30 %
Field experience (discussion forum)	0 - 10 %
Skills	15%

Online:

Assessment system	Weighting
On-campus knowledge tests	60%
Reports and papers	10 - 20 %
Alternative assessment methods	10 - 20 %
Field experience (discussion forum)	0 - 5 %
Skills	10 - 20 %

On the Campus Virtual, when you open the subject area, you will find all the details of your assessable tasks and the deadlines and assessment procedures for each task.

7.1. Ordinary exam period

To pass the subject area in the ordinary exam period, you will need a final grade of at least 5.0 out of 10.0 for the subject area.

The weighted average will be used provided that:

• You achieve a grade of at least 5 in the knowledge tests.

• You achieve a grade of at least 5 in the case study/problem scenario.

• You achieve a grade of at least 5 in the average of the individual activities/

laboratories / portfolio.

You must have a minimum of 50% attendance in the on-campus lectures. The attendance for seminars or events online will be determined by the professor.

7.2. Extraordinary exam period (resits)

To pass the subject area in the ordinary exam period, you will need a final grade of at least 5.0 out of 10.0 (weighted average) for the subject area.

In any case, you will need a grade of at least 5.0 in the final test for it to be included in the weighting with the other activities. The same applies to the activities that are specifically indicated to be passed, in order to be able to average the rest.

Activities not passed in the ordinary exam period, or those not submitted, must be submitted after receiving the relevant corrections and feedback from the lecturer. This can be changed by a personalised remediation plan agreed with the professor. In any case, the remediation of activities must be agreed with the professor at least one month before the end date of the extraordinary exam period.

8. TIMELINE

The timeline with submission dates for the assessable tasks in this subject area will be indicated in this section:

Assessable tasks	Date
Activities UA1 (individual)	Week 3
Activities UA2 (individual)	Week 6
Activities UA3 (individual)	Week 9
Activities UA4 (individual)	Week 12
Activity UA5 (individual)	Week 13
Activity UA5 (in groups)	Week 15

The timeline may be subject to change for logistical reasons related to the activities. Students will be informed of any changes in due time and course.

9. BIBLIOGRAPHY

• Eckel, Bruce (2008). "Thinking in Java" 4th ed. Prentice Hall.

• Horstmann, C.S. (2018). "Core Java I – Fundamentals" 11th ed. Prentice Hall.

• Horstmann, C.S. (2016). "Core Java II – Advanced Features" 10th ed. Prentice Hall.

• Schildt, H. (2018). "Java. A Beginner's Guide" 8th ed. Oracle Press.

• Schildt, H. (2018). "Java. The Complete Reference" 11 th ed. Oracle Press.

10. DIVERSITY AWARENESS UNIT

Students with special educational needs:

To ensure equal opportunities, curricular adaptations or adjustments for students with special educational needs will be outlined by the Diversity Awareness Unit (UAD, Spanish acronym).

As an essential requirement, students with special educational needs must obtain a report about the curricular adaptations/adjustments from the Diversity Awareness Unit by contacting unidad.diversidad@universidadeuropea.es at the beginning of each semester.

11. STUDENT SATISFACTION SURVEYS

Your opinion matters!

Universidad Europea encourages you to complete our satisfaction surveys to identify strengths and areas for improvement for staff, degrees and the learning process.

These surveys will be available in the survey area of your campus virtual or by email.

Your opinion is essential to improve the quality of the degree.

Many thanks for taking part.