1. OVERVIEW

Subject area	Fundamentals of Programming
Degree	Bachelor's Degree in Data Science
School/Faculty	Faculty of Science, Engineering and Design
Year	First
ECTS	6 ECTS
Type	Core
Language(s)	Spanish
Delivery Mode	On campus
Semester	First semester

2. INTRODUCTION

Fundamentals of Programming is one of the basic subject areas in the Degree in Data Science teaching syllabus at the Universidad Europea. This subject introduces students to basic concepts on how to approach and solve problems through algorithms, as well as how to apply these algorithms in programming language.

In this subject, students will learn about structured and sequential programming, as well as the concept of fundamental data structures and how to translate them to a specific programming language (in this case C++).

This conceptual knowledge of structured and sequential programming can easily be transferred to almost any other modern programming language - not just to the language studied in this subject.

Finally, students will learn how to approach a problem based on the input and output data, design algorithms to process the input data and obtain the required output data as well as write, analyse and debug these algorithms into a formal programming language.

3. SKILLS AND LEARNING OUTCOMES

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

- CB1: Students have shown their 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

Cross-curricular skills (CT, by the acronym in Spanish):

- CT2: Independent learning: skills for choosing strategies to search, analyse, evaluate and manage information from different sources, as well as to independently learn and put into practice what has been learnt.
- CT4: Written/spoken communication: ability to communicate and gather information, ideas, opinions and viewpoints to understand and be able to act, spoken through words or gestures or written through words and/or graphic elements.
- CT5 - Analysis and problem-solving: be able to critically assess information, break down complex situations, identify patterns and consider different alternatives, approaches and perspectives in order to find the best solutions and effective negotiations.

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

- CE3: Knowledge of the core principles and applications of software development and databases.
- CE4: Ability to successfully apply data type models and algorithms to create solutions to problems in the data science field.
- CE5: Ability to design, implement, gather, store and exploit databases and database management systems to create solutions to problems in the data science field.

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

- RA1: Develop algorithmic thinking. Transfer a problem into a sequence of actions to solve it.
- RA2: Design and implement solutions to problems of medium-level complexity using databases (structured, semi-structured, non-structured), data structures and object-oriented programming.
- RA3: Use programming environments to compile, link and execute programs, as well as identify and correct errors in each stage.
- RA4: Suitably document the designs, as well as the introduction of comments in the code to ease understanding and further use of the software created.

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

Skills	Learning outcomes
CB1, CB2, CT2, CT5, CE3	RA1
CB1, CB2, CT2, CT5, CE3, CE4	RA2
CB1, CB2, CT3, CT4, CT5, CE4, CE5	RA3, RA4

4. CONTENTS

1. Introduction to computing and programming.
2. Fundamentals of program design and mechanisation of abstraction.
3. Introduction to databases.
4. Data types and structures.
5. Basic constructions.
6. Structured programming

5. TEACHING/LEARNING METHODS

The types of teaching/learning methods are as follows:

- Collaborative learning: students learn to work with other people (colleagues and professors) to find creative, comprehensive and constructive solutions to questions and problems that arise from the given case studies, using relevant knowledge and available resources in relation to each subject.
- Problem-based learning: students face problems they must solve either working as a team or individually.
- Master Lecture: presentations by the professor using the appropriate technological tools to facilitate understanding of the subject matter.
- Workshop teaching: students acquire knowledge working with the instruments they will use in their future profession (learning by doing).
- Directed academic activities: more independent tasks (individual or in groups), involving search for information, written summaries, debates and public defence of work.

6. LEARNING ACTIVITIES

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

On campus:

Learning activity	Number of hours
Master lectures	31
Problem-solving	18
Case studies and field studies	10
Laboratory work	20
Debates and discussions	5
Learning contract (definition of interests, needs and objectives)	2
Autonomous learning	54
Tutorials	9
On campus knowledge tests	2
TOTAL	**150**

7. ASSESSMENT

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

On campus:

Assessment system	Weighting
On campus knowledge tests, either theoretical or practical	60

Off-site tests to assess theory/practical learning	25
Assessment of attitude	5
Self- and co-assessment	5
Laboratory, workshop or simulation tests	5

On the Virtual Campus, when you open the subject area, you can see all the details of your assessment activities and the deadlines and assessment procedures for each activity.

8. BIBLIOGRAFÍA

- Javier Ceballos Sierra, C-C++ Curso de Programación-Ra-Ma (2007)
- Walter Savitch, Absolute C++, Pearson Addison-Wesley, 2006, 2nd edition (2009)
- Practicas online: https://www.w3schools.com/CPP/cpp intro.asp
- Miguel Ángel Acera García, C/C++. Curso de programación (Manuales Imprescindibles) (2017)