

**1. General information****Course:** PROGRAMMING FUNDAMENTALS I**Type:** BASIC**Degree:** 405 - DEGREE IN COMPUTER SCIENCE ENGINEERING (TA)**Center:** 15 - FACULTY OF SOCIAL SCIENCES AND INFORMATION TECHNOLOGIES**Year:** 1**Main language:** Spanish**Use of additional languages:****Web site:** <https://campusvirtual.uclm.es>**Code:** 42302**ECTS credits:** 6**Academic year:** 2020-21**Group(s):** 60**Duration:** First semester**Second language:** Spanish**English Friendly:** Y**Bilingual:** N**Lecturer:** MANUEL DE CASTRO VÁZQUEZ - Group(s): 60

Building/Office	Department	Phone number	Email	Office hours
Pool / 1.2	TECNOLOGÍAS Y SISTEMAS DE INFORMACIÓN		Manuel.deCastro@uclm.es	

**2. Pre-Requisites**

Since this is a subject of the first semester of the first year of the degree in Computer Engineering, and will also be new for most students, it does not seem logical that prior requirements should be established.

In spite of this, and with the main objective of guaranteeing the comprehension of the contents and the acquisition of skills of this subject, it is advisable for the student to take advantage of certain personal skills and abilities that he acquired during his primary and secondary studies. Among them we will highlight the critical reading of the texts in the bibliography, the use of electronic material for this subject available on the Moodle platform and the active search for complementary material on the network.

It is also very important that the student has personal initiative and perseverance in the daily work of the subject.

**3. Justification in the curriculum, relation to other subjects and to the profession**

The subject Programming Fundamentals I, like the other subjects in the academic plan, is not an isolated subject but is a part of it with a close relationship to the rest of the subjects.

Within this whole, the subject has been included within the subject PROGRAMMING together with the subjects Programming Fundamentals II, Data Structure, Programming Methodology and Concurrent and Real Time Programming.

As it is the first subject of this module, the subject Programming Fundamentals I will be one of the fundamental pillars, since the basic concepts of programming that the rest of the subjects of the module will use later will be studied.

In addition, the knowledge and skills acquired with this subject will be important for the proper development of subjects from other subjects such as, for example, those related to Software Engineering.

If we go beyond the university context and think about the future incorporation of our students into the labour market, even if a graduate in computer science is not called to be a mere programmer but to be responsible for large projects, this subject, together with the rest of the subjects, will provide the necessary skills and abilities to be able to plan the project well and subsequently evaluate the different alternatives proposed.

**4. Degree competences achieved in this course****Course competences**

Code	Description
BA04	Basic knowledge about the uses and programming of computers, operating systems, data bases, and digital programmes with applications in engineering.
BA05	Knowledge about the structure, organization, functioning, and inter connexions of digital programmes, with their application in engineering problems.
CO07	Knowledge, design, and efficient use of types of data and structures which arise as most appropriate in problem solving.
CO08	Ability to analyse, design, build and maintain applications in a strong, safe, and efficient manner by selecting the most appropriate paradigms and programming languages.
INS01	Analysis, synthesis, and assessment skills.
INS04	Problem solving skills by the application of engineering techniques.
PER01	Team work abilities.
PER02	Ability to work in an international context.
PER04	Interpersonal relationship skills.
PER05	Acknowledgement of human diversity, equal rights, and cultural variety.
SIS01	Critical thinking.
SIS03	Autonomous learning.
UCLM02	Ability to use Information and Communication Technologies.

**5. Objectives or Learning Outcomes****Course learning outcomes**

## Description

Resolution of problems throughout basic techniques of algorithm design.

## 6. Units / Contents

**Unit 1: Algorithms**

**Unit 2: Control structures**

**Unit 3: Data structures**

**Unit 4: Modularity**

**Unit 5: Recursion**

## 7. Activities, Units/Modules and Methodology

Training Activity	Methodology	Related Competences (only degrees before RD 822/2021)	ECTS	Hours	As	Com	Description
Class Attendance (theory) [ON-SITE]	Lectures	BA04 BA05 CO07 CO08	0.72	18	N	-	Teaching of the subject matter by lecturer (MAG)
Individual tutoring sessions [ON-SITE]		BA04 BA05 CO07 CO08 UCLM02	0.18	4.5	N	-	Individual or small group tutoring in lecturer's office, classroom or laboratory (TUT)
Study and Exam Preparation [OFF-SITE]	Self-study	BA04 BA05 CO07 CO08 SIS01 SIS03	2.1	52.5	N	-	Self-study (EST)
Other off-site activity [OFF-SITE]	Practical or hands-on activities	BA04 BA05 CO07 CO08 INS01 INS04 PER01 PER02 PER04 PER05 SIS03	0.6	15	N	-	Lab practical preparation (PLAB)
Problem solving and/or case studies [ON-SITE]	Problem solving and exercises	BA04 BA05 CO07 CO08 INS04 PER01 PER02 PER04 PER05 SIS01 SIS03 UCLM02	0.6	15	Y	N	Worked example problems and cases resolution by the lecturer and the students (PRO)
Writing of reports or projects [OFF-SITE]	Self-study	BA04 BA05 CO07 CO08 INS01 INS04 PER02 PER04 PER05	0.9	22.5	Y	N	Preparation of essays on topics proposed by lecturer (RES)
Laboratory practice or sessions [ON-SITE]	Practical or hands-on activities	BA04 BA05 CO07 CO08 INS04 PER02 PER04 PER05	0.6	15	Y	Y	Realization of practicals in laboratory /computing room (LAB)
Final test [ON-SITE]	Assessment tests	BA04 BA05 CO07 CO08 INS01 INS04 PER01 PER02	0.3	7.5	Y	Y	According to the evaluation modality (EVA)
<b>Total:</b>			<b>6</b>	<b>150</b>			
<b>Total credits of in-class work: 2.4</b>			<b>Total class time hours: 60</b>				
<b>Total credits of out of class work: 3.6</b>			<b>Total hours of out of class work: 90</b>				

As: Assessable training activity

Com: Training activity of compulsory overcoming (It will be essential to overcome both continuous and non-continuous assessment).

## 8. Evaluation criteria and Grading System

Evaluation System	Continuous assessment	Non-continuous evaluation*	Description
Final test	50.00%	50.00%	Compulsory activity that can be retaken (rescheduling) to be carried out within the planned exam dates of the final exam call (convocatoria ordinaria).
Theoretical papers assessment	15.00%	15.00%	Non-compulsory activity that can be retaken. To be carried out before end of teaching period
Laboratory sessions	25.00%	25.00%	Compulsory activity that can be retaken. To be carried out during lab sessions
Oral presentations assessment	10.00%	10.00%	Non-compulsory activity that can be retaken. To be carried out during the theory/lab sessions by the continuous assesment students. Non-continuous evaluation students will be evaluated with an alternative system.
<b>Total:</b>	<b>100.00%</b>	<b>100.00%</b>	

According to art. 4 of the UCLM Student Evaluation Regulations, it must be provided to students who cannot regularly attend face-to-face training activities the passing of the subject, having the right (art. 12.2) to be globally graded, in 2 annual calls per subject, an ordinary and an extraordinary one (evaluating 100% of the competences).

### Evaluation criteria for the final exam:

#### Continuous assessment:

In compulsory activities, a minimum mark of 40% is required in order to pass that activity and have the possibility to therefore pass the entire subject. The evaluation of the activities will be global and therefore must be quantified by means of a single mark. If the activity consists of several sections, each section may be evaluated separately provided students are informed in writing of this evaluation criterion at the beginning of the academic year. In the case of the activities that may be retaken (i.e., rescheduling), an alternative activity or test will be offered in the resit/retake exam call (convocatoria extraordinaria). The final exam will be common for all the theory/laboratory groups of the subject and will be evaluated by the lecturers of the subject in a serial way, i.e., each part of the final exam will be evaluated by the same lecturer for all the students.

A student is considered to pass the subject if she/he obtains a minimum of 50 points out of 100, taking into account the points obtained in all the evaluable

activities, and also has passed all the recoverable activities, the student will have the opportunity to receive an alternative evaluation of those activities in the resit/retake exam call and, in that case, the final grade of the activity will correspond to the latter grade obtained.

The mark of the passed activities in any call, except for the final exam, will be conserved for the subsequent academic year at the request of the student, provided that mark is equal or greater than 50% and that the activities and evaluation criteria of the subject remain unchanged prior to the beginning of that academic year.

The failure of a student to attend the final exam will automatically result in her/him receiving a "Failure to attend" (no presentado). If the student has not passed any compulsory evaluation activity, the maximum final grade will be 40%.

#### Non-continuous evaluation:

Students who are unable to attend training activities on a regular basis may apply at the beginning of the semester for the non-continuous assessment mode. Similarly, if a student who is undergoing continuous assessment incurs any circumstance that prevents her/him from regularly attending the classroom-based training activities, she/he may renounce the accumulated mark in continuous assessment and apply for the non-continuous assessment mode. In this case, a notification by the student must be given before the date scheduled for the tests in the ordinary call, in accordance with a deadline that will be informed at the beginning of the semester.

Students who take the non-continuous assessment mode will be globally graded, in 2 annual calls per subject, an ordinary and an extraordinary one (evaluating 100% of the competences), through the assessment systems indicated in the column "Non-continuous assessment".

In the "non-continuous assessment" mode, it is not compulsory to keep the mark obtained by the student in the activities or tests (progress test or partial test) taken in the continuous assessment mode.

#### Specifications for the resit/retake exam:

Evaluation tests will be conducted for all recoverable activities.

#### Specifications for the second resit / retake exam:

Same characteristics as the resit/retake exam call.

9. Assignments, course calendar and important dates	
Not related to the syllabus/contents	
Hours	hours
<b>General comments about the planning:</b> The subject is taught in 3 x 1,5 hour sessions per week. The planning could be modified in the event of unforeseen causes.	

10. Bibliography and Sources						
Author(s)	Title/Link	Publishing house	Citv	ISBN	Year	Description
R. Sedgewick, K. Wayne	Introduction to Programming in Java: An Interdisciplinary Approach. 2nd Edition <a href="https://learning.oreilly.com/library/view/introduction-to-programming/9780134512389/">https://learning.oreilly.com/library/view/introduction-to-programming/9780134512389/</a>	Addison-Wesley		978-0672337840	2017	
H. Schildt	Java: A Beginner's Guide. 8th Edition. <a href="https://learning.oreilly.com/library/view/java-a-beginners/9781260440225/">https://learning.oreilly.com/library/view/java-a-beginners/9781260440225/</a>	McGraw-Hill		978-1260440218	2018	
Serrano Guerrero, Jesús; Vizcaíno Barceló, Aurora; Muñoz Caro, Camelia, Niño Ramos, Alfonso	Ejercicios resueltos de programación imperativa en Java <a href="http://www.bubok.es/libros/218626/Ejercicios-resueltos-de-programacion-imperativa-en-Java">http://www.bubok.es/libros/218626/Ejercicios-resueltos-de-programacion-imperativa-en-Java</a>	Bubok S.L.		978-84-686-2728-1	2012	Ejercicios resueltos en Java
Camelia Muñoz Caro, Alfonso Niño Ramos, Aurora Vizcaíno Barceló	Introducción a la programación con orientación a objetos	Prentice Hall		978-84-205-3440-4	2007	
Nair, Premchand S.	Java programming fundamentals: problem solving through objects	CRC Taylor & Francis		978-1-4200-6547-3	2009	