



## Teaching Guide

Identifying Data					2018/19
<b>Subject (*)</b>	Programming in C++	<b>Code</b>	614855232		
<b>Study programme</b>	Mestrado Universitario en Matemática Industrial (2013)				
Descriptors					
<b>Cycle</b>	<b>Period</b>	<b>Year</b>	<b>Type</b>	<b>Credits</b>	
Official Master's Degree	2nd four-month period	First	Optional	3	
<b>Language</b>					
<b>Teaching method</b>	Face-to-face				
<b>Prerequisites</b>					
<b>Department</b>	Matemáticas				
<b>Coordinador</b>	Ferreiro Ferreiro, Ana María	<b>E-mail</b>	ana.ferreiro@udc.es		
<b>Lecturers</b>	Ferreiro Ferreiro, Ana María García Rodríguez, José Antonio	<b>E-mail</b>	ana.ferreiro@udc.es jose.garcia.rodriguez@udc.es		
<b>Web</b>	sites.google.com/site/lep2cpp/				
<b>General description</b>	Los objetivos de la asignatura son dar a conocer los aspectos fundamentales de los lenguajes de programación C++, la programación orientada a objetos (POO) -- paradigma de programación dominante en el desarrollo de aplicaciones informáticas-- apoyándonos en C++ y su correspondiente aplicación en el desarrollo de aplicaciones informáticas del ámbito de la ingeniería,				

## Study programme competences

Code	Study programme competences

## Learning outcomes

Learning outcomes	Study programme competences		
Programación imperativa tradicional en C++			
Comprender los conceptos básicos POO (clases, objetos, etc), así como comprender las propiedades básicas de la POO (herencia, polimorfismo, sobrecarga, etc)			
Conocer las diferencias entre la programación imperativa tradicional y la programación orientada a objetos.			
Desarrollar código (implementado en C++) flexible y reutilizable apoyándonos en la POO.			

## Contents

Topic	Sub-topic
Tema 1: El lenguaje de programación C++	<ul style="list-style-type: none"> <li>- Introducción a la programación en C++</li> <li>- Tipos de datos básicos</li> <li>- I/O por teclado y por fichero</li> <li>- Sentencias de control</li> <li>- Gestión dinámica de memoria: punteros</li> <li>- Estructuras</li> <li>- Funciones. Sobrecarga</li> </ul>
Tema 2: Programación Orientada a Objetos en C++	<ul style="list-style-type: none"> <li>- Introducción a la Programación Orientada a Objetos</li> <li>- Clases e instancias</li> <li>- Sobrecarga de operadores</li> <li>- Funciones y clases friend</li> <li>- Herencia</li> <li>- Polimorfismo</li> <li>- Templates (plantillas)</li> </ul>



Tema 3: Standard Template Library (STL)	<ul style="list-style-type: none"> <li>- Introducción a la STL</li> <li>- Contenedores e iteradores</li> <li>- Manejo de contenedores básicos</li> </ul>
---	--

Planning				
Methodologies / tests	Competencies	Ordinary class hours	Student?s personal work hours	Total hours
Laboratory practice		18	0	18
Supervised projects		47	0	47
Guest lecture / keynote speech		10	0	10
Personalized attention		0	0	0

(\*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.

Methodologies	
Methodologies	Description
Laboratory practice	<p>Clases de prácticas tuteladas en las que los alumnos podrán en práctica mediante pequeños ejercicios los conceptos vistos en las clases teóricas.</p> <p>Se intercalarán las explicaciones teóricas con las prácticas, con el objetivo de facilitar el aprendizaje.</p>
Supervised projects	Proyectos y ejercicios a realizar por el alumno para profundizar en la comprensión de la materia
Guest lecture / keynote speech	En las clases teóricas se explicará la sintaxis del lenguaje de programación C++, se abordará la Programación Orientada a Objetos, así como la sintaxis para expresar los conceptos de la POO en C++

Personalized attention	
Methodologies	Description
Laboratory practice Guest lecture / keynote speech Supervised projects	En horas de tutoría y seguimiento de los trabajos prácticos que tienen que entregar los alumnos

Assessment			
Methodologies	Competencies	Description	Qualification
Supervised projects		<ul style="list-style-type: none"> <li>- Se propondrán trabajos semanalmente.</li> <li>- Se propondrá una práctica final, donde se apliquen todos los conceptos de POO estudiados en la asignatura</li> </ul>	100

Assessment comments
<ul style="list-style-type: none"> <li>- La evaluación se realizará sólo mediante diferentes trabajos prácticos y una práctica final, todos ellos de entrega obligatoria</li> <li>- Se propondrán trabajos semanalmente.</li> <li>- Se propondrá una práctica final, donde se apliquen todos los conceptos de POO estudiados en la asignatura.</li> <li>- Los trabajos semanales y la práctica final constituyen el 100% de la nota.</li> </ul>

Sources of information	
Basic	<ul style="list-style-type: none"> <li>- Ray Lischner (2003). C++ In a Nutshell. O'Reilly Media</li> <li>- Walter Savitch (2004). Problem Solving with C++: The Object of Programming, Fifth Edition. Addison-Wesley</li> <li>- Bjarne Stroustrup (2001 (2007 reimp.)). El Lenguaje de programación C++. Addison-Wesley Iberoamericana</li> <li>- Harvey M. Deitel, Paul J. Deitel (2009). C++ : cómo programar (6ª ed.). Pearson Educación</li> </ul>
Complementary	

Recommendations



Subjects that it is recommended to have taken before

Subjects that are recommended to be taken simultaneously

Subjects that continue the syllabus

Other comments

&lt;p&gt;Es

recomendable tener experiencia de programación en otros lenguajes de programación.&nbsp;&lt;/p&gt;

(\* )The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.