



Guía docente				
Datos Identificativos				2019/20
Asignatura (*)	Lenguajes y Entornos de Programación II	Código	614455209	
Titulación	Mestrado Universitario en Enxeñaría Matemática			
Descriptorios				
Ciclo	Periodo	Curso	Tipo	Créditos
Máster Oficial	2º cuatrimestre	Primero	Optativa	3
Idioma	Castellano			
Modalidad docente	Presencial			
Prerrequisitos				
Departamento	Matemáticas			
Coordinador/a		Correo electrónico		
Profesorado		Correo electrónico		
Web	sites.google.com/site/lep2cpp/			
Descripción general	Los objetivos de la asignatura son dar a conocer los aspectos fundamentales de los lenguajes de programación C y 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,			

Competencias / Resultados del título	
Código	Competencias / Resultados del título

Resultados de aprendizaje			
Resultados de aprendizaje			Competencias / Resultados del título
Programación imperativa tradicional en C/C++			
Conocer las diferencias entre la programación imperativa tradicional y la programación orientada a objetos.			
Comprender los conceptos básicos POO (clases, objetos, etc), así como comprender las propiedades básicas de la POO (herencia, polimorfismo, sobrecarga, etc)			
Desarrollar código (implementado en C++) flexible y reutilizable apoyándonos en la POO.			

Contenidos	
Tema	Subtema
Introducción a los lenguajes de programación C y C++	
El lenguaje de programación C	Tipos de datos básicos y derivados. Estructuras de control. Punteros, Funciones. Archivos de cabecera.
Mejoras no relacionadas con la POO introducidas por el lenguaje de programación C++	Declaración simplificada de variables de tipo enumeración y estructuras. E/S por consola. Lectura y Escritura de ficheros. Scope de las variables. Reserva dinámica de memoria. Funciones: valores por defecto, funciones inline, Variables de tipo referencia.
Introducción a la programación orientada a objetos (POO)	Objetos. Clases. Encapsulación. Objetos frente a clases



Programación orientada a objetos en C++	Clases y objetos. Jerarquía de clases y herencia. Polimorfismo. Sobrecarga de operadores. Métodos virtuales. Templates.
---	--

Planificación				
Metodologías / pruebas	Competencias / Resultados	Horas lectivas (presenciales y virtuales)	Horas trabajo autónomo	Horas totales
Sesión magistral		10	0	10
Prácticas de laboratorio		18	0	18
Trabajos tutelados		47	0	47
Atención personalizada		0		0

(\*) Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de los alumnos

Metodologías	
Metodologías	Descripción
Sesión magistral	En las clases teóricas se explicará la sintaxis del lenguaje de programación C/C++, se abordará la Programación Orientada a Objetos, así como la sintaxis para expresar los conceptos de la POO en C++
Prácticas de laboratorio	Clases de prácticas tuteladas en las que los alumnos podrán en práctica mediante pequeños ejercicios los conceptos vistos en las clases teóricas.  Se intencarán las explicaciones teóricas con las prácticas, con el objetivo de facilitar el aprendizaje.
Trabajos tutelados	Proyectos y ejercicios a realizar individualmente por el alumno para profundizar en la comprensión de la materia.

Atención personalizada	
Metodologías	Descripción
Sesión magistral Prácticas de laboratorio	

Evaluación			
Metodologías	Competencias / Resultados	Descripción	Calificación
Trabajos tutelados		Cada alumno tendrá que presentar dos proyectos en el que se abordarán aspectos relacionados con los contenidos desarrollados a lo largo del curso.	100

Observaciones evaluación
A lo largo del curso cada alumno tendrá que presentar dos proyectos, donde se aborden la mayor parte de los contenidos desarrollados a lo largo del curso. Estos trabajos supondrán el 100% de la nota final.

Fuentes de información
------------------------



<b>Básica</b>	<ul style="list-style-type: none"> <li>- B.W. Kernighan, D.M. Ritchie (1991). El lenguaje de programación C (2ª ed.). Prentice Hall</li> <li>- Harvey M. Deitel, Paul J. Deitel (2009). C++ : cómo programar (6ª ed.). Pearson Educación</li> <li>- Bjarne Stroustrup ([2001] (2007 reimp.)). El Lenguaje de programación C++ . Addison-Wesley Iberoamericana,</li> <li>- Bruce Eckel (2000). Thinking in C++: Introduction to Standard C++, Volume One (2nd Edition) (Vol 1). Disponible Online</li> <li>- Bruce Eckel (2003). Thinking in C++: Introduction to Standard C++, Volume One (2nd Edition) (Vol 2). Disponible Online</li> </ul> <p>Los libros de Bruce Eckel pueden descargarse de forma gratuita de la web:  <a href="http://mindview.net/Books/TICPP/ThinkingInCPP2e.html">http://mindview.net/Books/TICPP/ThinkingInCPP2e.html</a> Los libros de Bruce Eckel pueden descargarse de forma gratuita de la web: <a href="http://mindview.net/Books/TICPP/ThinkingInCPP2e.html">http://mindview.net/Books/TICPP/ThinkingInCPP2e.html</a></p>
<b>Complementaria</b>	<ul style="list-style-type: none"> <li>- Scott Meyers (2005). Effective C++ 55 specific ways to improve your programs and designs (3ª ed). Addison-Wesley,</li> <li>- Walter Savitch (2004). Problem Solving with C++: The Object of Programming, Fifth Edition . Addison-Wesley</li> <li>- British Standards Institute (2005). C++ estándar. Anaya. Serie de programación</li> <li>- J.D. García, J.M. Pérez, L.M. Sánchez, J. Carretero, F. García (2004). Problemas resueltos de Programación en Lenguaje C++. Thomson</li> </ul>

### Recomendaciones

#### Asignaturas que se recomienda haber cursado previamente

#### Asignaturas que se recomienda cursar simultáneamente

#### Asignaturas que continúan el temario

Lenguajes y Entornos de Programación I/614455104

#### Otros comentarios

Es recomendable tener experiencia de programación en otros lenguajes de programación y en particular haber cursado la asignatura de "Lenguajes y entornos de programación I".

Al alumno debe orientar el estudio de la asignatura en el ámbito de la práctica, puesto que la mejor técnica para adquirir destreza de cualquier lenguaje de programación es la práctica del mismo. Por ello se recomienda completar el mayor número posible de las prácticas propuestas y también experimentar por cuenta propia.

(\*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías