



GUÍA DOCENTE
Curso 2011-2012

Titulación:	Grado en Matemáticas			Código :	701G
Centro:	FCEAI				
Dirección:	Edificio CCT C / Madre de Dios, 51			Código postal:	26006
Teléfono:	+34 941 299607	Fax:	+34 941299611	Correo electrónico:	decanato.cai@unirioja.es
Director del Grado:	Judith Mínguez Cenicerros				
Teléfono:	+34 941 299466	Correo electrónico:	direstudios.matematicas@unirioja.es		
Despacho:	219	Edificio:	Vives		

Fdo.:Judith Mínguez Cenicerros

En Logroño, a 1 de julio de 2011

GUÍA DOCENTE

Curso 2011-2012

Titulación:	Grado en Matemáticas			701G	
Asignatura:	Especificación y desarrollo de sistemas de software			701209013	
Materia:					
Módulo:	M3 Programación				
Carácter:	Obligatorio	Curso:	2º	Semestre:	2º
Créditos ECTS:	6	Horas presenciales:	60	Horas de trabajo autónomo estimadas:	90
Idiomas en los que se imparte:	castellano				
Idiomas del material de lectura o audiovisual:	castellano, inglés				

Departamentos responsables de la docencia:

Matemáticas y Computación	R111
Dirección:	Luis de Ulloa, s/n
Código postal:	26004
Teléfono:	+34 941 299452
Fax:	+34 941 299460
Correo electrónico:	dpto.dmc@unirioja.es

Profesores

Profesor responsable de la asignatura:	Ana Romero Ibáñez		
Teléfono:	+34 941 299438	Correo electrónico:	
Despacho:	235	Edificio:	Vives
Horario de tutorías:			
Nombre profesor:	Laureano Lambán Pardo		
Teléfono:	+34 941 299446	Correo electrónico:	lalamban@unirioja.es
Despacho:	223	Edificio:	Vives
Horario de tutorías:			

Descripción de contenidos:

- Abstracción de datos. Especificación /implementación de tipos de datos. Relación con la orientación a objetos. Aplicaciones al diseño orientado a objetos.
- Especificación de algoritmos.
- Conceptos sobre lenguajes: sintaxis y semántica.
- Análisis de algoritmos: eficiencia y corrección.

Requisitos previos:

Se aconseja tener conocimientos y competencias relacionadas con la programación, especialmente con los fundamentos básicos del paradigma orientado a objetos.

Relación de asignaturas que proporcionan los conocimientos y competencias requeridos:

Metodología de la Programación
Tecnología de la Programación
Programación orientada a objetos

Contexto

La asignatura pertenece al módulo "Informática" y aparece en el segundo semestre del segundo curso. Dentro de dicho bloque, es la continuación natural de la asignatura "Programación Orientada a Objetos". Se presupone a los alumnos cierta familiaridad con nociones básicas sobre programación orientada a objetos (clase, objeto, herencia, etc.). Por su parte, pretende revisar algunos conceptos básicos de programación desde un punto de vista más abstracto.

Dentro del módulo "Informática", la asignatura pretende dotar a los alumnos de las competencias necesarias para ser capaces de diseñar e implementar aplicaciones desde el paradigma orientado a objetos, viendo los elementos esenciales de la orientación a objetos como herramientas necesarias para dicho diseño. El otro punto esencial de la asignatura es aportar una perspectiva formal (mayor nivel de abstracción) sobre diferentes aspectos relacionados con la programación (sintaxis, semántica, corrección y eficiencia), buscando una mejora en los hábitos del alumno a la hora de programar, que mejore la calidad y fiabilidad de su trabajo.

Las prácticas de laboratorio de la asignatura consisten en el diseño y desarrollo (en un lenguaje que permita orientación a objetos) de los sistemas de clases necesarios para la construcción de pequeñas aplicaciones. Se pretende que, a lo largo de la asignatura, aparezcan involucrados en las prácticas los elementos centrales de la programación orientada a objetos, vistos aquí como elementos necesarios para el diseño.

Competencias:**Competencias generales**

- **CG6:** Relacionar el conocimiento especializado de Matemáticas con el conocimiento general en el que se inserta y con las herramientas que utiliza cuando se aplica en diversas opciones profesionales, especialmente en el marco de las TIC.
- **CG7:** Saber abstraer las propiedades estructurales de objetos de la realidad observada y de otros ámbitos, distinguiéndolas de aquellas puramente ocasionales, comprobando la aplicabilidad de las Matemáticas.
- **CG8:** Capacitar para el aprendizaje autónomo de nuevos conocimientos y técnicas.

Competencias específicas

- **CE4:** Encontrar soluciones algorítmicas de problemas matemáticos y de aplicación (de ámbito académico, técnico, financiero o social), sabiendo comparar distintas alternativas, según criterios de adecuación, complejidad y coste.
- **CE5:** Saber programar algoritmos de modo correcto y eficaz, eligiendo convenientemente lenguajes y plataformas de programación.

Resultados del aprendizaje:

- Aplicar los elementos característicos de la programación orientada a objetos en el diseño y la programación de

- aplicaciones.
- Utilizar los conceptos relacionados con la abstracción de datos para facilitar el desarrollo de aplicaciones desde un enfoque orientado a objetos.
 - Comprender y aprender a establecer descripciones formales de algoritmos.
 - Conocer algunos conceptos básicos sobre sintaxis y semántica de lenguajes de programación.
 - Aprender a realizar pequeñas verificaciones formales de corrección de algoritmos, conocimiento enfocado a adquirir mejores hábitos de programación.
 - Conocer y aplicar conceptos y herramientas relacionados con el análisis de la eficiencia.
 - Usar con solvencia algún lenguaje que permita un enfoque orientado a objetos.

Temario

1. Introducción: "Abstracción y formalismo en Programación".
2. Especificación/Implementación de tipos de datos.
3. Tipos abstractos y orientación a objetos.
4. Especificación de algoritmos.
5. Sintaxis y semántica de un lenguaje imperativo sencillo.
6. Nociones sobre eficiencia. Verificación formal de la corrección de algoritmos.

Bibliografía

A. Tucker, R. Noonan.

"Lenguajes de programación: principios y paradigmas".

Mc Graw Hill, 2003.

R. D. Tennent.

"Specifying software".

Cambridge University Press, 2002.

R. Peña.

"Diseño de programas: formalismo y abstracción".

Prentice Hall, 1997.

C. Muñoz, A. Niño, A. Vizcaíno.

"Introducción a la programación con orientación a objetos".

Prentice Hall, 2002.

C.T. Wu.

"Introducción a la programación orientada a objetos con Java".

Mc Graw Hill, 2001.

J. Bishop.

"Java: Fundamentos de programación".

Addison-Wesley, 1999.

T. Budd.

"Object-oriented programming".

Addison-Wesley, 2002.

F. Gutiérrez , F. Durán, E. Pimentel.

“Programación orientada a objetos con Java”

Thomson, 2007.

J. Lewis, J. Chase.

“Estructuras de datos con Java. Diseño de estructuras y algoritmos”.

Pearson Addison Wesley, 2006.

Página oficial de Java.

java.sun.com

Metodología

Modalidades organizativas:	Métodos de enseñanza:
- MO1: Clases teóricas - MO2: Seminarios y talleres - MO3: Clases prácticas - MO5: Tutorías - MO6: Estudio y trabajo autónomo del alumno	- ME1: Lección magistral - ME2: Aprendizaje basado en problemas - ME3: Resolución de ejercicios y problemas - ME4: Utilización de recursos informáticos

Organización

Actividades presenciales:	Horas
- Clases teóricas	30
- Clases prácticas de aula	28
- Pruebas presenciales de evaluación	2
- Otras actividades	
Total horas presenciales	60
Actividades no presenciales (trabajo autónomo):	Horas estimadas
- Estudio autónomo individual o en grupo	25
- Resolución individual de ejercicios, cuestiones u otros trabajos, actividades en biblioteca o similar	50
- Preparación en grupo de trabajos, presentaciones (orales, debates,...), actividades en biblioteca o similar	15
Total horas estimadas de trabajo autónomo	90
Total horas	150

Evaluación

Sistemas de evaluación:	% sobre total	Recuperable/ No Recuperable
<i>Examen final</i>	70	Rec.
<i>Prácticas en grupo.</i>	10	Rec.
<i>Prácticas individuales.</i>	20	No Rec.

Comentario:

Para los estudiantes a tiempo parcial (reconocidos como tales por la Universidad), las actividades de evaluación no recuperable podrán ser sustituidas por otras, a especificar en cada caso. Esta posibilidad se habilitará siempre y cuando la causa que le impida la realización de la actividad de evaluación programada sea la que ha llevado al reconocimiento de la dedicación a tiempo parcial.

Criterios críticos para superar la asignatura: