

Facultad de Ciencias

## GUÍA DOCENTE DE LA ASIGNATURA

G2005 - Métodos de Programación

Grado en Matemáticas  
Optativa. Curso 4

Curso Académico 2021-2022

### 1. DATOS IDENTIFICATIVOS

Título/s	Grado en Matemáticas		Tipología v Curso	Optativa. Curso 4	
Centro	Facultad de Ciencias				
Módulo / materia	MENCIÓN EN INFORMÁTICA				
Código y denominación	G2005 - Métodos de Programación				
Créditos ECTS	6	Cuatrimestre	Cuatrimestral (1)		
Web					
Idioma de impartición	Español	English friendly	Sí	Forma de impartición	Presencial

Departamento	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA				
Profesor responsable	MARIO ALDEA RIVAS				
E-mail	mario.aldea@unican.es				
Número despacho	Facultad de Ciencias. Planta: + 3. DESPACHO - COMPUTADORES TIEMPO REAL (3054)				
Otros profesores	ADOLFO GARANDAL MARTIN				

### 2. CONOCIMIENTOS PREVIOS

Conocimientos básicos de programación (preferentemente en lenguajes Java o C).

### 3. COMPETENCIAS GENÉRICAS Y ESPECÍFICAS DEL PLAN DE ESTUDIOS TRABAJADAS

Competencias Genéricas
(Autonomía) Aprender de manera autónoma nuevos conocimientos y técnicas.
(Trabajar en equipo) Saber trabajar en equipo.
(Leer) Leer textos científicos escritos tanto en español como en inglés.
Competencias Específicas
(Modelizar) Proponer, analizar, validar e interpretar modelos de situaciones reales sencillas, utilizando las herramientas matemáticas más adecuadas a los fines que se persigan.
(Resolver) Resolver problemas de Matemáticas, mediante habilidades de cálculo básico y otros, planificando su resolución en función de las herramientas de que se disponga y de las restricciones de tiempo y recursos.
(Utilizar software) Utilizar aplicaciones informáticas de análisis estadístico, cálculo numérico y simbólico, visualización gráfica, optimización u otras para experimentar en Matemáticas y resolver problemas.
(Desarrollar programas) Desarrollar programas que resuelvan problemas matemáticos utilizando para cada caso el entorno computacional adecuado.
Competencias Básicas
Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

#### 3.1 RESULTADOS DE APRENDIZAJE

- Conocer los fundamentos de la programación de computadores y del razonamiento sobre los programas, incluyendo modularidad y orientación a objetos.
- Ser capaz de argumentar convincentemente la corrección de estructuras de bucle y recursivas individuales en base únicamente al análisis estático.
- Ser capaz de aplicar con corrección y eficacia criterios de descomposición modular de problemas en grado suficiente como para completar la programación de su solución.
- Conocer con familiaridad y ser capaz de emplear correcta y eficazmente las nociones fundamentales de la programación orientada a objetos, incluyendo diagramas de clases.
- Ser capaz de realizar entrada/salida de información sobre ficheros.
- Ser capaz de documentar correctamente el código fuente de un programa.

#### 4. OBJETIVOS

El principal objetivo de la asignatura es que el alumno domine los principales conceptos de la metodología de programación orientada a objetos y sepa aplicarles al desarrollo de programas de mediana complejidad utilizando el lenguaje Java.

La asignatura pretende dotar al alumno de los siguientes conocimientos:

- Conocimientos básicos sobre el proceso integral de desarrollo de software.
- Diferencia y relación entre los conceptos de clase, objeto y referencia.
- Conceptos de clase y tipo primitivo.
- Utilización de las clases como elemento básico de estructuración del código.
- Conceptos de herencia y polimorfismo.
- Tratamiento de errores mediante excepciones.
- Modularidad y la abstracción como principios básicos para la construcción de una aplicación.
- Utilización de tablas y secuencias: recorridos y búsquedas.
- Diferencia entre ficheros binarios y de texto.
- Técnicas básicas para la prueba de módulos.

La asignatura utilizará el lenguaje Java como lenguaje de programación sobre el que concretar y practicar los conceptos estudiados.

#### 5. MODALIDADES ORGANIZATIVAS Y MÉTODOS DOCENTES

ACTIVIDADES	HORAS DE LA ASIGNATURA
<b>ACTIVIDADES PRESENCIALES</b>	
HORAS DE CLASE (A)	
- Teoría (TE)	20
- Prácticas en Aula (PA)	10
- Prácticas de Laboratorio Experimental(PLE)	
- Prácticas de Laboratorio en Ordenador (PLO)	30
- Prácticas Clínicas (CL)	
Subtotal horas de clase	60
<b>ACTIVIDADES DE SEGUIMIENTO (B)</b>	
- Tutorías (TU)	7,5
- Evaluación (EV)	7,5
Subtotal actividades de seguimiento	15
<b>Total actividades presenciales (A+B)</b>	<b>75</b>
<b>ACTIVIDADES NO PRESENCIALES</b>	
Trabajo en grupo (TG)	20
Trabajo autónomo (TA)	55
Tutorías No Presenciales (TU-NP)	
Evaluación No Presencial (EV-NP)	
<b>Total actividades no presenciales</b>	<b>75</b>
<b>HORAS TOTALES</b>	<b>150</b>

## 6. ORGANIZACIÓN DOCENTE

CONTENIDOS		TE	PA	PLE	PLO	CL	TU	EV	TG	TA	TU- NP	EV- NP	Semana
1	TEMA 1. Desarrollo de software orientado a objetos  1.1. Ingeniería del software 1.2. Procesos de desarrollo de software 1.3. Introducción al análisis de requisitos 1.4. Diseño orientado a objetos 1.5. Lenguaje Unificado de Modelado (UML) 1.6. Programación orientada a objetos 1.7. Proceso de desarrollo usado en la asignatura	4,00	0,00	0,00	4,00	0,00	1,00	1,00	3,00	8,00	0,00	0,00	1,2
2	Tema 2. Programación orientada a objetos en Java  2.1. Visión general del lenguaje Java 2.2. Clases y objetos en Java 2.3. Comparación con C 2.4. Creación e inicialización de objetos 2.5. Tipos primitivos, referencias y objetos 2.6. Comparación de objetos 2.7. Recolector de basura 2.8. Métodos y campos de clase (o estáticos)	5,00	3,00	0,00	6,00	0,00	1,50	1,50	4,00	11,00	0,00	0,00	3,4,5
3	Tema 3. Diseño Modular  3.1. Modularidad y abstracción 3.2. Modificadores de acceso básicos 3.3. Diseño modular: cohesión y acoplamiento 3.4. Paquetes 3.5. Programación con módulos predefinidos 3.6. Documentación del código fuente 3.7. Patrón Modelo-Vista-Controlador	3,00	2,00	0,00	6,00	0,00	1,50	1,00	4,00	9,00	0,00	0,00	6,7,8
4	TEMA 4. Tratamiento de Errores  4.1. Introducción 4.2. Excepciones Java: generación y propagación automática 4.3. Bloques de tratamiento excepciones 4.4. La cláusula finally 4.5. Patrones de tratamiento de excepciones 4.6. Lanzar excepciones 4.7. Creación de excepciones propias 4.8. Excepciones "comprobadas" y cláusula throws 4.9. Notificación de errores mediante excepciones 4.10. Usos incorrectos de las excepciones	3,00	2,00	0,00	6,00	0,00	1,50	1,50	3,00	10,00	0,00	0,00	9,10,11
5	Tema 5. Herencia y Polimorfismo  5.1. Herencia 5.2. Clases abstractas 5.3. Polimorfismo 5.4. La clase Object	3,00	2,00	0,00	6,00	0,00	1,50	1,50	4,00	11,00	0,00	0,00	12,13,14

6	TEMA 6. Entrada/salida con ficheros	2,00	1,00	0,00	2,00	0,00	0,50	1,00	2,00	6,00	0,00	0,00	15
	6.1 Ficheros												
	6.2 Flujos de datos (streams)												
	6.3 Escritura de ficheros de texto												
	6.4 Lectura de ficheros de texto												
	6.5 Escritura de ficheros binarios												
	6.6 Lectura de ficheros binarios												
	6.7 Ficheros binarios de objetos												
	6.8 Resumen tipos de ficheros												
TOTAL DE HORAS		20,00	10,00	0,00	30,00	0,00	7,50	7,50	20,00	55,00	0,00	0,00	
Esta organización tiene carácter orientativo.													

Ante la situación incierta de que las medidas de distanciamiento social establecidas por las autoridades sanitarias no permitan desarrollar alguna actividad docente de forma presencial en el aula para todos los estudiantes matriculados, se adoptará una modalidad mixta de docencia que combine esta docencia presencial en el aula con docencia a distancia. De la misma manera, la tutorización podrá ser sustituida por tutorización a distancia utilizando medios telemáticos.

TE	Horas de teoría
PA	Horas de prácticas en aula
PLE	Horas de prácticas de laboratorio experimental
PLO	Horas de prácticas de laboratorio en ordenador
CL	Horas de prácticas clínicas
TU	Horas de tutoría
EV	Horas de evaluación
TG	Horas de trabajo en grupo
TA	Horas de trabajo autónomo
TU-NP	Tutorías No Presenciales
EV-NP	Evaluación No Presencial

## 7. MÉTODOS DE LA EVALUACIÓN

Descripción	Tipología	Eval. Final	Recuper.	%
Prácticas	Evaluación en laboratorio	No	Sí	15,00
Calif. mínima	0,00			
Duración	2 horas			
Fecha realización	A lo largo del curso			
Condiciones recuperación	Examen en la convocatoria extraordinaria			
Observaciones	<p>Las prácticas deberán entregarse dentro del plazo establecido. Una entrega fuera del plazo supondrá una penalización en la nota.</p> <p>La nota se obtiene en base a:</p> <ul style="list-style-type: none"> <li>- Evaluación del material entregado por el alumno (código, diagrama de clases, etc.)</li> <li>- Presentación de las prácticas: a lo largo del curso el profesor podrá requerir al alumno la presentación de alguna de las prácticas realizadas.</li> </ul>			
Exámenes parciales	Evaluación en laboratorio	No	Sí	85,00
Calif. mínima	4,50			
Duración	Variable (entre 1 y 3 horas)			
Fecha realización	A lo largo del curso			
Condiciones recuperación	Examen en la convocatoria extraordinaria			
Observaciones	<p>Evaluación de los conocimientos prácticos y teóricos del alumno. Se celebrarán entre 3 y 5 exámenes parciales a lo largo del curso.</p> <p>La nota de este apartado se calculará como la media (ponderada) de las notas de los exámenes realizados.</p> <p>La nota de los exámenes parciales podrá verse mejorada si el alumno realiza satisfactoriamente los problemas y ejercicios que podrían proponerse a lo largo del curso.</p>			
<b>TOTAL</b>				<b>100,00</b>
<b>Observaciones</b>				
Para poder optar a superar la asignatura es necesario entregar, al menos, el 75% de las prácticas de la asignatura.				
<b>Criterios de evaluación para estudiantes a tiempo parcial</b>				
<p>Los estudiantes a tiempo parcial que no puedan realizar alguno de los exámenes parciales, podrán recuperar dichos exámenes en el examen del periodo oficial y/o de la convocatoria extraordinaria.</p> <p>Se espera de estos alumnos que vayan realizando las prácticas de la asignatura al ritmo que se van proponiendo y que las vayan entregando en los plazos que a tal efecto se establezcan. La entrega fuera de plazo de las prácticas les supondrá la misma penalización que a los alumnos que asisten habitualmente a las clases de la asignatura.</p> <p>Para poder optar a superar la asignatura es necesario entregar, al menos, el 75% de las prácticas de la asignatura.</p>				

## 8. BIBLIOGRAFÍA Y MATERIALES DIDÁCTICOS

### BÁSICA

Ken Arnold, James Gosling, David Holmes, "El lenguaje de programación Java", 3a edición. Addison-Wesley, 2000.

Francisco Gutiérrez, Francisco Durán, Ernesto Pimentel. "Programación Orientada a Objetos con Java". Paraninfo, 2007.

Complementaria
D.J. Barnes y M. Kölling, "Objects first with Java: A practical introduction using Bluej", Prentice Hall, 2003.
Ian Sommerville, "Ingeniería de software" (9a edición). Pearson Educación de México, 2011.
King, Kim N. "Java programming: from the beginning". W. W. Norton & Company, cop. 2000
The Java Tutorials. <a href="http://docs.oracle.com/javase/tutorial/">http://docs.oracle.com/javase/tutorial/</a>
Eitel, Harvey M. y Deitel, Paul J., "Cómo programar en Java", 9a edición. Pearson Educación, Mexico, 2012.

## 9. SOFTWARE

PROGRAMA / APLICACIÓN	CENTRO	PLANTA	SALA	HORARIO
Entorno de desarrollo Eclipse	Facultad de Ciencias			
Sistema operativo Linux	Facultad de Ciencias			
Java Development Kit	Facultad de Ciencias			

## 10. COMPETENCIAS LINGÜÍSTICAS

- Comprensión escrita                       Comprensión oral  
 Expresión escrita                             Expresión oral  
 Asignatura íntegramente desarrollada en inglés

### Observaciones