GUIA DOCENTE ABREVIADA DE LA ASIGNATURA

G2005 - Métodos de Programación

Grado en Matemáticas

Curso Académico 2026 - 2027

1. DATOS IDENTIFICATIVOS					
Título/s	Grado en Matemáticas			Tipología y 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

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.

6. ORGANIZACIÓN DOCENTE

	CONTENIDOS
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
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)
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
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

5	Tema 5. Herencia y Polimorfismo 5.1. Herencia 5.2. Clases abstractas 5.3. Polimorfismo 5.4. La clase Object
6	TEMA 6. Entrada/salida con ficheros 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

7. MÉTODOS DE LA EVALUACIÓN

Descripción	Tipología	Eval. Final	Recuper.	%
Prácticas	Evaluación en laboratorio	No	Sí	15,00
Exámenes parciales	Evaluación en laboratorio	No	Sí	85,00
Examen final (alumnos tiempo parcial)	Evaluación en laboratorio	Sí	Sí	0,00
TOTAL				100,00

Observaciones

Para poder optar a superar la asignatura es necesario entregar, al menos, el 75% de las prácticas de la asignatura.

Criterios de evaluación para estudiantes a tiempo parcial

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.
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.
Para poder optar a superar la asignatura es necesario entregar, al menos, el 75% de las prácticas de la asignatura.

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.

Esta es la Guía Docente abreviada de la asignatura. Tienes también publicada en la Web la información más detallada de la asignatura en la Guía Docente Completa.