

GUÍA DOCENTE

DATOS GENERALES DE LA ASIGNATURA

ASIGNATURA:	Programación II
PLAN DE ESTUDIOS:	Grado en Ingeniería Informática
FACULTAD:	Escuela Politécnica Superior
CARÁCTER DE LA ASIGNATURA:	Básica
ECTS:	6 créditos
CURSO:	Primero
SEMESTRE:	Segundo
IDIOMA EN QUE SE IMPARTE:	Castellano
PROFESORADO:	Juan Jesús Tortajada Cordero
DIRECCIÓN DE CORREO ELECTRÓNICO:	juan.tortajada@uneatlantico.es

DATOS ESPECÍFICOS DE LA ASIGNATURA

REQUISITOS PREVIOS:
No Aplica
CONTENIDOS:
<p>Presentación de la Asignatura</p> <p>Capítulo 1. Programación Orientada a Objetos, programación de eventos, herencia y polimorfismo.</p> <ol style="list-style-type: none"> 1.1. Programación Orientada a Objetos (POO). Relaciones de asociación. 1.2. Relaciones de asociación. Modelado de problemas. 1.3. Relaciones de asociación. Implementación de problemas. 1.4. Introducción a la programación en el entorno integrado de desarrollo

- 1.5. Introducción a la programación dirigida por eventos.
- 1.6. Programación visual.
- 1.7. Biblioteca de componentes visuales
- 1.8. Programación Orientada a Objetos. Herencia, modelación e implementación de problemas.
- 1.9. Relaciones de asociación y herencia. Sistematización.
- 1.10. Programación Orientada a Objetos. Polimorfismo. Modelado e implementación de problemas.
- 1.11. Herencia múltiple.
- 1.12. Lenguaje Unificado de Modelado (UML)

Capítulo 2. Tratamiento de errores y excepciones

- 2.1. Introducción.
- 2.2. Modelar problemas utilizando el tratamiento de errores y excepciones.

Capítulo 3. Genericidad

- 3.1. Caracterizar la genericidad y su utilidad en el desarrollo POO.
- 3.2. Modelar problemas utilizando la genericidad.
- 3.3 Implementar soluciones a problemas haciendo uso de la genericidad.

COMPETENCIAS

COMPETENCIAS GENERALES:

- Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes.
- Capacidad de explicar y aplicar las materias básicas y tecnologías, que permitan el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
- Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
- Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio
- 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 tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética
- Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
- Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

COMPETENCIAS ESPECÍFICAS:

- Conocer y aplicar la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y saber aplicarlos en la resolución de problemas propios de la ingeniería.
- Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- Capacidad de explicar y aplicar los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- Capacidad de entender y utilizar de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.

RESULTADOS DE APRENDIZAJE:

- Conocer los fundamentos de un lenguaje de programación de alto nivel
- Conocer los principales algoritmos para resolución de problemas comunes
- Saber manejar los tipos de datos, estructuras de datos - lineales y no lineales - y tipos abstractos de datos de forma correcta y adecuada
- Comprender y saber utilizar eficientemente la orientación a objetos y sus conceptos derivados: herencia, polimorfismo, abstracción

METODOLOGÍAS DOCENTES Y ACTIVIDADES FORMATIVAS

METODOLOGÍAS DOCENTES:

En esta asignatura se ponen en práctica diferentes metodologías docentes con el objetivo de que los alumnos puedan obtener los resultados de aprendizaje definidos anteriormente:

- MD1 Método expositivo
- MD2 Estudio y análisis de casos
- MD3 Resolución de ejercicios
- MD4 Aprendizaje basado en problemas
- MD5 Aprendizaje orientado a proyectos
- MD6 Aprendizaje cooperativo / Trabajo en grupos
- MD7 Trabajo autónomo

ACTIVIDADES FORMATIVAS:

A partir de las metodologías docentes especificadas anteriormente, en esta asignatura, el alumno participará en las siguientes actividades formativas:

Actividades formativas	
Actividades dirigidas	Clases expositivas
	Clases prácticas
Actividades supervisadas	Supervisión de actividades
	Tutorías (individual / en grupo)
Actividades autónomas	Preparación de clases
	Estudio personal y lecturas
	Elaboración de trabajos
	Trabajo individual en campus virtual

El primer día de clase, el profesor/a proporcionará información más detallada al respecto.

SISTEMA DE EVALUACIÓN

CONVOCATORIA ORDINARIA:

En la convocatoria ordinaria de esta asignatura se aplican los siguientes instrumentos de evaluación:

	Actividades de evaluación	Ponderación
Evaluación continua	1 Exámen Parcial	25 %
	Entregas de Portfolios y Ejercicios	20 %
	Interés y participación del alumno en la asignatura	5 %
Evaluación final	Examen Teórico-Práctico	50 %

CONVOCATORIA EXTRAORDINARIA:

La convocatoria extraordinaria tendrá lugar durante el mes de julio (consúltese el calendario académico fijado por la universidad). Esta consistirá en la realización de uno o dos Exámenes Teórico-Prácticos con un valor de hasta el 50% de la nota final de la asignatura. El resto de la nota se complementará con la calificación obtenida en la evaluación continua de la convocatoria ordinaria.

BIBLIOGRAFÍA Y RECURSOS DE REFERENCIA GENERALES

BIBLIOGRAFÍA BÁSICA:

- Fundamentos de Programación con Java (Serbat, Andrés; Gonzalez, Patricia)

- Guide to Java, A concise introduction to programming (James T. Streib, Takako Soma)
- Introduction to programming in Java, An interdisciplinary approach (Robert Sedgewick, Kevin Wayne)

BIBLIOGRAFÍA COMPLEMENTARIA:

- *Algorithms, 4th Edition* (Robert Sedgewick, Kevin Wayne)

WEBS DE REFERENCIA:

- <http://www.java.com/es/>
- <http://docs.oracle.com/javase/8/>
- <https://developers.google.com/blockly/>
- <http://introcs.cs.princeton.edu/java/home/>
- <https://atom.io/>
- <http://algs4.cs.princeton.edu/home/>

OTRAS FUENTES DE CONSULTA:

No Aplica