

DATOS BÁSICOS DE LA GUÍA DOCENTE:

Materia:	PROGRAMACIÓN ORIENTADA A OBJETOS I		
Identificador:	33416		
Titulación:	DOBLE GRADO EN INGENIERÍA INFORMÁTICA Y BIOINFORMÁTICA		
Módulo:	PROGRAMACION Y DESARROLLO		
Tipo:	OBLIGATORIA		
Curso:	2	Periodo lectivo:	Primer Cuatrimestre
Créditos:	6	Horas totales:	150
Actividades Presenciales:	65	Trabajo Autónomo:	85
Idioma Principal:	Castellano	Idioma Secundario:	Inglés
Profesor:	ARCEGA RODRIGUEZ, LORENA (T)	Correo electrónico:	larcega@usj.es

PRESENTACIÓN:

En la asignatura Programación Orientada a Objetos se hace una introducción al paradigma de la programación orientada a objetos, incidiendo en los conceptos fundamentales: Clase, Herencia y Polimorfismo. Se incide en la creación de software robusto haciendo uso del tratamiento de excepciones y en los conceptos básicos de análisis y diseño utilizando lenguajes de modelado.

Para profundizar en la técnica de la Programación Orientada a Objeto se utilizan lenguajes de modelado y programación vigentes en las empresas de desarrollo de software y videojuegos.

COMPETENCIAS PROFESIONALES A DESARROLLAR EN LA MATERIA:

Competencias Generales de la titulación	G02	Capacidad innovadora para proponer y encontrar formas nuevas y eficaces de realizar cualquier tarea y/ o función dentro de su entorno profesional con una elevada motivación por la calidad
	G05	Capacidad para adaptarse a diferentes entornos con una actitud positiva y optimista y orientar su conducta a la consecución de metas
	G06	Capacidad para analizar y resolver los problemas o imprevistos complejos que puedan surgir durante la actividad profesional dentro de cualquier tipo de organización socio-económica
	G08	Habilidad para comunicarse eficazmente sobre distintos temas en una variedad de contextos profesionales y con los diferentes medios disponibles
	G09	Capacidad para tomar decisiones de manera imparcial y desde un punto de vista racional
	G10	Capacidad crítica y analítica en la evaluación de información, datos y líneas de actuación
	G13	Capacidad para utilizar estrategias de aprendizaje individuales orientadas a la mejora continua en el ejercicio profesional y para emprender estudios posteriores de forma autónoma
	G14	Capacidad de abstracción para manejar diferentes modelos complejos de conocimiento y aplicarlos al planteamiento y resolución de problemas
	Competencias Específicas de la titulación	E03
E04		Capacidad para mantener una actitud abierta a la innovación y la creatividad en el marco de la profesión de ingeniería
E08		Capacidad para comunicarse productivamente con clientes, usuarios y colegas, tanto de modo oral como por escrito, con el fin de transmitir ideas, resolver conflictos y alcanzar consensos
E09		Capacidad para mantener las competencias profesionales mediante el aprendizaje autónomo y la mejora continua
E11		Capacidad para mantenerse al día en el mundo tecnológico y empresarial en el ámbito de las tecnologías de la informática y comunicaciones
E12		Capacidad para gestionar la complejidad a través de la abstracción, el modelado, las "best practices", los patrones, los estándares y el uso de herramientas apropiadas
E13		Capacidad para identificar, evaluar y usar tecnologías actuales y emergentes, considerando su aplicabilidad en función de las necesidades de individuos y organizaciones
E16		Capacidad para comprender un dominio de aplicación hasta el punto de ser capaz de desarrollar aplicaciones IT adecuadas para el mismo
E17		Capacidad para identificar y analizar las necesidades de los usuarios con el objetivo de diseñar soluciones IT efectivas y usables que puedan integrarse en el entorno operativo del

		usuario.
	E18	Capacidad para identificar y definir los requisitos que deben ser satisfechos por los sistemas IT para satisfacer las necesidades planteadas por organizaciones o individuos
	E20	Capacidad para realizar el diseño detallado de los componentes del proyecto (procedimientos, interfaz de usuario, características de equipos, parámetros de los sistemas de comunicaciones, etc.).
	E27	Capacidad para elaborar y mantener documentación descriptiva de la génesis, producción y operatividad de los sistemas informáticos

REQUISITOS PREVIOS:

Se debe haber cursado la asignatura de primer curso Fundamentos de Programación o equivalente

PROGRAMACIÓN DE LA MATERIA:

Contenidos de la materia:

1 - UML
1.1 - Introducción a la POO
1.2 - Modelado de Software
1.3 - Introducción UML
1.4 - Casos de uso
1.5 - Diagramas de clase
1.6 - Diagramas de interacción
1.7 - Diagramas de objeto
1.8 - Diagramas de paquete
1.9 - Diagramas de actividad
1.10 - Diagramas de estado
1.11 - Diagramas de componente
1.12 - Diagramas de despliegue
2 - Lenguaje C plus plus
2.1 - Introducción C plus plus
2.2 - Clases
2.3 - Sobrecarga de operadores
2.4 - Herencia y polimorfismo
2.5 - Plantillas
2.6 - Excepciones
2.7 - STL

La planificación de la asignatura podrá verse modificada por motivos imprevistos (rendimiento del grupo, disponibilidad de recursos, modificaciones en el calendario académico, etc.) y por tanto no deberá considerarse como definitiva y cerrada.

METODOLOGÍAS Y ACTIVIDADES DE ENSEÑANZA Y APRENDIZAJE:

Metodologías de enseñanza-aprendizaje a desarrollar:

- Sesiones teóricas

El profesor expondrá apoyado en recursos TIC (portátil, proyector, Internet) y la pizarra la parte teórica de la asignatura, principalmente las dudas que hayan surgido a los estudiantes durante la realización del trabajo autónomo. El material utilizado en cada sesión estará siempre en la PDU con antelación para que los alumnos puedan realizar una lectura previa. Se aconseja encarecidamente a los alumnos leer con anterioridad los temas a tratar en clase.

Se valorará la participación, debate, preguntas e inquietudes mostradas por los alumnos

relacionadas con la asignatura, tanto en las sesiones presenciales como en las que se desarrollen fuera de las sesiones presenciales

- Sesiones prácticas - Aprendizaje basado en resolución de ejercicios y problemas

Cada tema irá acompañado de un boletín de problemas. Los más representativos serán resueltos en clase. Aquellas dudas que surjan a los alumnos sobre los no resueltos en clase también serán tratadas en clase.

Con anterioridad a la realización de las pruebas escritas se facilitará a los estudiantes un ejemplo de examen que será resuelto con anterioridad a la prueba escrita.

- Aprendizaje basado en resolución de proyecto

Una parte muy importante del aprendizaje de la asignatura, y de la nota, se conseguirá con la resolución de una práctica a resolver de manera individual por los alumnos. Se proporcionará a los alumnos los enunciados de las prácticas y se solicitará su entrega en un tiempo prudencial. Se realizarán entregas parciales de la práctica. La resolución correcta de esta práctica preparará al alumno para adquirir con una gran eficiencia las competencias propias de la asignatura.

- Sesiones de tutoría

Los estudiantes podrán preguntar al profesor aquellas dudas que no han podido ser resueltas durante las clases o que hayan surgido en su tiempo dedicado al trabajo autónomo. Se podrá solicitar bibliografía de ampliación específica de algún tema concreto y/ o cualquier otro tipo de información relacionada con la asignatura. Por otra parte, durante estas sesiones, se realizará un seguimiento de supervisión y orientación del proceso a seguir en cada una de las actividades realizadas.

Las tutorías se fijarán de mutuo acuerdo con el profesor y estudiantes en fechas próximas a la entrega/ publicación de esta guía docente.

Volumen de trabajo del alumno:

Modalidad organizativa	Métodos de enseñanza	Horas estimadas
Actividades Presenciales	Clase magistral	23
	Resolución de prácticas, problemas, ejercicios etc.	20
	Exposiciones de trabajos de los alumnos	2
	Prácticas de laboratorio	12
	Actividades de evaluación	8
Trabajo Autónomo	Asistencia a tutorías	4
	Estudio individual	23
	Preparación de trabajos individuales	40
	Tareas de investigación y búsqueda de información	6
	Lecturas obligatorias	8
	Lectura libre	2
	Otras actividades de trabajo autónomo	2
Horas totales:		150

SISTEMA DE EVALUACIÓN:

Obtención de la nota final:

Pruebas escritas:	50	%
Trabajos individuales:	40	%
Participación y evaluación continua:	10	%
TOTAL	100	%

*Las observaciones específicas sobre el sistema de evaluación serán comunicadas por escrito a los



alumnos al inicio de la materia.

BIBLIOGRAFÍA Y DOCUMENTACIÓN:

Bibliografía básica:

Ivar Jacobson, James Rumbaugh, Grady Brooch, Pack UML, ISBN: 8478290443
Bjarne Stroustrup, Título: El Lenguaje de Programación C , ISBN: 847829046X

Bibliografía recomendada:

Joyanes Aguilar, Luis. Programación en C : algoritmos, estructuras de datos y objetos, ISBN: 844814645X
Mari Carmen Juan Lizamora; José Hernández Orallo; Enrique Hernández Orallo; Mari Carmen Juan Lizamora; José Hernández Orallo; Enrique Hernández Orallo, C Estándar, ISBN: 8497320409
Josuttis, Nicolai M, The C Standard Library: A Tutorial and Reference
Stevens, Perdita; Pooley, R. J., (aut.), Joyanes Aguilar, Luis; Fernández Alarcón, Marta; González Crespo, Rubén, (tr.) Utilización de uml 2ED Editorial: Pearson Addison-Wesley ISBN: 8478290869 ISBN-13: 9788478290864
Martin Fowler y Kendall Scott, UML gota a gota Editorial: Pearson Addison Wesley ISBN: 9684443641
Deitel, H. M., Deitel, P. J. , Cómo Programar en C Editorial: Prentice Hall ISBN: 9702602548
Bruce Eckel, Thinking in C (Volumen 1 y 2) Ed: Prentice Hall ISBN: 0130353132 ISBN-13: 9780130353139 ISBN: 0139798099 ISBN-13: 9780139798092

Páginas web recomendadas:

Recursos C	http://www.cplusplus.com/
Curso de POO	http://www.jlmarina.net/metodos2/
Herramientas UML	http://www.objectsbydesign.com/tools/umltools_byPrice.html
BOUML: Herramienta UML	http://bouml.free.fr/documentation.html
Curso de C	http://c.conclase.net/curso/index.php?cap=000
Curso de C	http://www.zator.com/Cpp/E_Ce.htm