



**GUÍA DOCENTE DE**  
**Programación Orientada a Objetos**  
Curso 2011-2012

TITULACION: GRADO EN INGENIERÍA DE COMPUTADORES

GUIA DOCENTE DE LA ASIGNATURA: PROGRAMACIÓN ORIENTADA A OBJETOS

Profesores

Nombre y apellidos: Carlos Lázaro Carrascosa, Isidoro Hernán Losada

Coordinador/a de la asignatura: Carlos Lázaro Carrascosa

#### I.- Identificación de la asignatura

Tipo	OBLIGATORIA
Materia	PROGRAMACIÓN ORIENTADA A OBJETOS
Período de impartición	SEGUNDO CURSO, PRIMER SEMESTRE
Nº Créditos	6
Idioma en el que se imparte	CASTELLANO
Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS I
Asignaturas llave	
Tasa de éxito	

## II.- Presentación

El objetivo de la asignatura es presentar fundamentos de programación y diseño orientados a objetos, usando el lenguaje Java. La asignatura aborda la enseñanza de la orientación a objetos tanto desde el punto de vista del diseño como desde el de la implementación. Al final de la asignatura, el alumnado deberá ser capaz de:

- Conocer y Comprender los conceptos básicos de programación orientada a objetos.
- Conocer y Comprender los elementos esenciales del lenguaje Java
- Codificar sencillos programas orientados a objetos en el lenguaje Java.
- Conocer y Comprender conceptos básicos de diseño de programas orientados a objetos.
- Diseñar aplicaciones pequeñas en Java usando el lenguaje UML.

(\*Requisitos previos\*) Como requisito previo para cursar esta asignatura se necesitarán los conocimientos y competencias relacionados con la asignatura Introducción a la Programación detallados más adelante.

(\*Recomendaciones\*) Se recomienda al alumno haber superado con éxito la asignatura Estructuras de Datos.

(\*Conocimientos y competencias\*) Como requisito previo para cursar esta asignatura se necesitarán los siguientes conocimientos y competencias relacionados con la asignatura Introducción a la Programación:

G5. Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 3.2.

G8.- Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones

G9.- 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

B4. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería

C6. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

C7. Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema



C8. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados

E1. Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática

Conocimientos previos:

2.4.1. Conocerá las componentes principales del desarrollo de programas.

2.4.2. Será capaz de interpretar y utilizar conceptos relacionados con los programas y la programación.

2.4.3. Será capaz de resolver problemas con iniciativa, toma de decisiones y creatividad.

2.4.4. Será capaz de organizar y planificar el trabajo.

2.4.5. Aprenderá a trabajar en grupo.

2.4.6. Aprenderá a adquirir conocimientos de forma autónoma.

2.4.7. Mejorará su capacidad de análisis y de síntesis.

2.4.8. Será capaz de comunicar de forma efectiva conocimientos, técnicas, resultados e ideas relacionados con el contenido de la materia estudiada

### III.- Competencias

Competencias transversales	
G5	Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 3.2
G6	Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 3.2
G8	Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones
G9	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
<b>Competencias específicas (Comunes a la rama de Informática, de Formación básica y de Tecnología específica)</b>	
B4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
B5	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
C6	Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
C7	Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.



---

C8	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
C16	Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.

#### IV.- Contenido

##### IV. A. Temario de la asignatura

Bloque temático	Tema	Apartados
I.- "Introducción"	Tema 1. "Introducción"	Motivación de la POO. Conceptos básicos de programación orientada a objetos: Clases, objetos y referencias. Métodos y mensajes. Atributos y estado. Constructores. Vectores. El lenguaje Java: historia, sintaxis básica, entornos de programación.
II.- "Introducción al diseño OO"	Tema 2. "Principios de diseño orientado a objetos"	Diagramas de Clases UML. Otros Diagramas UML. Problemas sencillos de diseño.
III.- "Conceptos de Orientación a Objetos"	Tema 3. "Elementos de Java"	Clases e Interfaces. Paquetes. Atributos y Métodos de Instancia y de Clase (Estáticos). Atributos finales. Accesibilidad. Clases de Utilidad (Cadenas, Envoltorios).
	Tema 4. "Herencia y Polimorfismo"	Concepto y mecanismos de herencia. Herencia en Interfaces. Enlace dinámico: Polimorfismo.
	Tema 5. "Aplicaciones orientadas a objetos"	La biblioteca de APIs de Java: Colecciones, Genéricos, Entrada/Salida, Excepciones. Asociaciones entre clases. Introducción al diseño de interfaces gráficas y a la programación basada en eventos. Enumerados.

#### IV. B. Actividades obligatorias (evaluables):

##### 1. Prácticas

###### Prácticas obligatorias

Se entregarán dos prácticas obligatorias realizadas fuera del aula y en los laboratorios a lo largo del cuatrimestre. Ambas serán **acumulativas** y se harán en **grupos de no más de tres integrantes**. Son **no-reevaluables** debido a que se deben hacer a lo largo del curso, introduciendo en momentos concretos los conceptos adecuados. El contenido de las prácticas estará relacionado con el momento del curso en el que se hagan, y se trata de prácticas supervisadas, en las que la orientación del profesor es importante. Tienen un carácter local, cobran su sentido sólo si se hacen en el momento adecuado.

La primera se centrará en comprobar que el alumno comprende y sabe diseñar correctamente una aplicación.

La segunda se centrará en comprobar que el alumno comprende y sabe implementar correctamente la aplicación diseñada en la primera práctica.

Con ambas prácticas se comprobará que el alumno comprende y sabe emplear con el dominio suficiente **todos** los conocimientos adquiridos durante el curso.

El material a entregar como resultado del trabajo realizado en todas ellas será un **diseño** y un **programa** que cumpla las especificaciones indicadas en cada uno de los enunciados de las prácticas. A este material se le podrá añadir (a criterio del profesorado) una **memoria** descriptiva del trabajo realizado. La estructura y contenidos de la memoria, si procede, se indicarán en el enunciado.

Ambas prácticas se calificarán de acuerdo al material entregado y a la defensa presencial de dichas prácticas.

##### 2. Laboratorios

Se entregarán prácticas obligatorias realizadas en los laboratorios a lo largo del cuatrimestre (entre una y tres). Todas ellas serán **acumulativas**. El contenido de las prácticas estará relacionado con el momento del curso en el que se hagan, y se trata de prácticas supervisadas y a realizar en el laboratorio, en los que la presencia del profesor es importante. Es decir, tienen un carácter local, cobran su sentido sólo si se hacen en el momento adecuado. Por ello se trata de prácticas **no reevaluables**, aunque totalmente compensables con los demás elementos de evaluación del curso. Todas ellas se realizarán **individualmente**.

El material a entregar como resultado del trabajo realizado en todas ellas será un **programa** que cumpla las especificaciones indicadas en cada uno de los enunciados de las prácticas. A este material se le podrá añadir (a criterio del profesorado) una **memoria** descriptiva del trabajo realizado. La estructura y contenidos de la memoria, si procede, se indicarán en el enunciado.

El profesor corregirá las prácticas y publicará las calificaciones y, opcionalmente, podrá requerir la

defensa presencial e individual de las prácticas de aquellos estudiantes que estime oportuno.

### 3. Otras

Se propondrán algunos temas relacionados con Java y con la POO de carácter más avanzado, para que el alumnado realice un breve trabajo de revisión. Estos trabajos se realizarán en **grupos** de no más de tres personas, se expondrán en clase y tienen el carácter de **acumulativos**. Son **no reevaluables**, debido a que la exposición debe hacerse durante el período lectivo y delante de todo el grupo, aunque son totalmente compensables con el resto de los elementos de evaluación del curso.

Se realizarán pruebas (una o dos) de preguntas cortas o test, preferentemente a través del campus virtual. El contenido de las pruebas estará relacionado con el momento del curso en el que se hagan, y se trata de pruebas a realizar en el laboratorio, en los que la presencia del profesor es importante. Es decir, tienen un carácter local, cobran su sentido sólo si se hacen en el momento adecuado. Por ello se trata de prácticas **no reevaluables**, aunque totalmente compensables con los demás elementos de evaluación del curso. Estas pruebas tienen el carácter de **acumulativas e individuales**.

Se realizará un examen global a final de curso, individual y por escrito, que podrá incluir preguntas de teoría y problemas relacionados con el temario de todo el curso. Será necesario obtener al menos un cinco (sobre diez) en esta prueba, que es **reevaluable**.



#### V.- Tiempo de trabajo

Clases teóricas	16
Clases prácticas/de resolución de problemas, casos, etc.	8
Prácticas en laboratorios tecnológicos, clínicos, etc.	24
Realización de pruebas	8
Tutorías académicas	9
Actividades relacionadas: jornadas, seminarios, etc.	0
Preparación de clases teóricas	25
Preparación de clases prácticas/problemas/casos	45
Preparación de pruebas	15
<b>Total de horas de trabajo del estudiante</b>	<b>150</b>

## VI.- Metodología y plan de trabajo

### Clases teóricas

Periodo	Contenidos
Semanas 1 a 3	Tema 1. "Introducción"
Semanas 4 y 5	Tema 2. "Principios de diseño orientado a objetos"
Semanas 6 y 7	Tema 3: "Elementos de Java"
Semana 8	Tema 4: "Herencia y Polimorfismo":
Semanas 9 a 13	Tema 5. "Aplicaciones orientadas a objetos II":

### Prácticas de resolución de problemas, casos, etc.

Periodo	Contenidos
Semanas 1 a 3	Tema 1. "Introducción"
Semanas 4 y 5	Tema 2. "Principios de diseño orientado a objetos"
Semanas 6 y 7	Tema 3: "Elementos de Java"
Semana 8	Tema 4: "Herencia y Polimorfismo":
Semanas 9 a 13	Tema 5. "Aplicaciones orientadas a objetos II":

### Laboratorios

Periodo	Contenidos
Semanas 1 a 3	Prácticas de laboratorio correspondientes al tema 1, "Introducción"
Semana 4	Práctica de laboratorio correspondientes al tema 2, "Principios de diseño OO"
Semana 5	Práctica obligatoria, primera parte
Semanas 6 y 7	Práctica de laboratorio correspondientes al tema 3, "Elementos de Java"
Semana 8	Práctica de laboratorio correspondientes al tema 4, "Herencia y Polimorfismo"
Semanas 9 y 10	Prácticas de laboratorio correspondientes al tema 5, "Aplicaciones orientadas a objetos II":



---

Semana 11 y 12	Práctica obligatoria, segunda parte
Semana 13	Exposición de trabajos de ampliación

#### Seminarios y trabajos colectivos

Periodo	Contenidos
Semana 13	Defensa de la práctica final

#### Tutorías académicas

Periodo	
Semanas 1-13	1 hora semanal (a demanda) donde se resuelven dudas de los contenidos impartidos previamente

#### Pruebas

Fecha	Contenidos
Diciembre	Examen final

## VII.- Métodos de evaluación

### VII. A. Ponderación para la evaluación continua

El sistema de evaluación de las titulaciones en el marco del EEES es la evaluación continua.

En el sistema de evaluación continua la asistencia a clase es obligatoria y su valoración en el proceso de evaluación continua de la asignatura la establecerán los profesores en cada asignatura.

**% Mínimo de asistencia a clase: 80%.**

Actividad evaluadora	Tipo <sup>1</sup>		Ponderación	Periodo	Contenido
Prueba:	<input checked="" type="checkbox"/> Acumulativa		10%	Semana 6-9	El impartido hasta la fecha
<input checked="" type="checkbox"/> Test	<input type="checkbox"/> Liberatoria Puntuación mínima (de 1 a 10):.....	<input type="checkbox"/> Reevaluable <input checked="" type="checkbox"/> No reevaluable			
Prueba:	<input type="checkbox"/> Acumulativa		40%	Diciembre	Todo
<input checked="" type="checkbox"/> Preguntas de desarrollo escritas	<input checked="" type="checkbox"/> Liberatoria Puntuación mínima (de 1 a 10): 5	<input checked="" type="checkbox"/> Reevaluable <input type="checkbox"/> No reevaluable			
Prácticas dentro del aula:	<input checked="" type="checkbox"/> Acumulativa		10%	Durante el curso	El impartido hasta la fecha
<input checked="" type="checkbox"/> Resolución de problemas	<input type="checkbox"/> Liberatoria Puntuación mínima (de 1 a 10):.....	<input type="checkbox"/> Reevaluable <input checked="" type="checkbox"/> No reevaluable			
Prácticas dentro del aula:	<input checked="" type="checkbox"/> Acumulativa		10%	Al final del curso	Trabajos de ampliación
	<input type="checkbox"/> Liberatoria	<input type="checkbox"/> Reevaluable			

<sup>1</sup> Cada una de las actividades evaluables pueden tener una calificación liberatoria o acumulativa para la calificación final. Se indicará, si hay una puntuación mínima exigida a las pruebas para que se consideren aprobadas y sean liberatorias. Se especificará si las pruebas son orales o escritas, y si son o no reevaluables.



<input checked="" type="checkbox"/> Presentación de trabajos en grupo	Puntuación mínima (de 1 a 10):.....	<input checked="" type="checkbox"/> No reevaluable			
Prácticas fuera del aula:	<input checked="" type="checkbox"/> Acumulativa		30%	A lo largo del curso	10% diseño, 15% implementación, 5% defensa
<input checked="" type="checkbox"/> Trabajos de grupo	<input type="checkbox"/> Liberatoria Puntuación mínima (de 1 a 10):.....	<input type="checkbox"/> Reevaluable <input checked="" type="checkbox"/> No reevaluable			
<b>Total</b>			<b>100%</b>		

### VII. B. Ponderación para la evaluación de alumnos a tiempo parcial

Para que un alumno pueda optar a esta evaluación, tendrá que obtener la "Dispensa Académica" para la asignatura, que habrá solicitado al Decano o Director/a del Centro que imparte su titulación.

La "Dispensa Académica" no excluye de la evaluación continua. Dicha evaluación se acomodará por el profesor, asistido por el coordinador de grado, estableciéndose la adaptación curricular según las características de cada caso concreto.

En esta asignatura, la dispensa académica supondrá la adecuación del calendario (si fuese necesario) a las necesidades particulares de los alumnos acogidos a dicha modalidad. Lo que, en la práctica, significa que el alumno con dispensa académica se someterá a las mismas pruebas que los alumnos sometidos a evaluación continua pero en plazos distintos.

### VII. C. Revisión de las pruebas de evaluación.

Un elemento fundamental del nuevo sistema de enseñanza-aprendizaje y de la evaluación continua es el relativo a la retroalimentación permanente que reciben los estudiantes sobre el trabajo realizado. Esto se traduce en la necesidad de devolverles información precisa, objetiva e inmediata acerca de sus resultados de rendimiento, de manera que puedan monitorizar sus niveles de esfuerzo y regular su trabajo, al mismo tiempo que dicha información actúa como un elemento motivador.

Con este objetivo, el profesorado de esta asignatura publicará las calificaciones de cada elemento de evaluación en campus virtual en fecha cercana a su entrega, siempre que sea posible. Generalmente, esta calificación irá acompañada de comentarios respecto a la solución de cada alumno. Los alumnos podrán ponerse en contacto con el profesor para resolver las posibles dudas que surjan del proceso. Además, al final de cada convocatoria se establecerá una fecha para realizar una revisión global.

### VIII.- Recursos y materiales didácticos

#### General

Título	Programación orientada a objetos con Java
Autor	Francisco Durán, Francisco Gutiérrez, Ernesto Pimentel
Editorial	Thomson 2007
Título	Programación II, Teoría y práctica del módulo de programación orientada a objetos
Autor	L. Fernández Muñoz, F. Arroyo Montoro
Editorial	Departamento de publicaciones de la EUI de Madrid 1997
Título	Análisis y diseño estructurado y orientado a objetos de sistemas informáticos
Autor	De Amescua Seco y otros
Editorial	McGraw Hill. 2003

#### Complementaria

Título	Fundamentos de programación. Algoritmos, Estructuras de datos y Objetos
Autor	L. Joyanes Aguilar
Editorial	McGraw-Hill 2003

#### Direcciones web de interés

Dirección 1: - Programación en Java: Fundamentos de programación y principios de diseño. Fernando Berzal. Universidad de Granada <a href="http://elvex.ugr.es/decsai/java/">http://elvex.ugr.es/decsai/java/</a>
Dirección 2:- API Java 6. <a href="http://download.oracle.com/javase/6/docs/api/overview-summary.html">http://download.oracle.com/javase/6/docs/api/overview-summary.html</a>

IX.- Profesorado

Nombre y apellidos	Carlos Lázaro Carrascosa
Horario de tutorías académicas	Se informará al principio del semestre en clase y a través del campus virtual
Correo electrónico	Campus Virtual
Departamento/área de conocimiento	Lenguajes y Sistemas Informáticos I / Lenguajes y Sistemas Informáticos
Categoría	Profesor Colaborador
Titulación Académica	Licenciado en Informática
Experiencia Docente (Indicar la antigüedad en el área y en la asignatura. Incluir tramos de docencia.)	10 años en el área, 5 años en la asignatura, 1 tramo docente.  Experiencia en diferentes asignaturas de programación: Metodología y Tecnología de la programación, Estructuras de datos y de la información, Estructuras de datos y Algoritmos Avanzados, Diseño y Análisis de Algoritmos.
Experiencia profesional (Indicar la actividad profesional y la antigüedad en la misma)	10 años como profesor en la Universidad Rey Juan Carlos, realizando diferentes labores docentes, investigadoras y de gestión.

Nombre y apellidos	Isidoro Hernán Losada
Horario de tutorías académicas	Por determinar
Correo electrónico	isidoro.hernan@urjc.es
Departamento/área de conocimiento	Lenguajes y Sistemas Informáticos I
Categoría	Profesor Colaborador
Titulación Académica	Licenciado en CC. Físicas Especialidad de Cálculo Automático por la UCM
Experiencia Docente	Profesor Asociado a tiempo parcial del Departamento de Lenguajes y Sistemas Informáticos de la Universidad Complutense de Madrid (1999-2000). Asignatura: Metodología de la Programación.  Profesor de la URJC desde Octubre-2000, de distintas asignaturas de programación:



	<p>Metodología y Tecnología de la Programación de ITIG e ITIS</p> <p>Programación Orientada a Objetos de I. Telecomunicaciones.</p> <p>Introducción a la Programación de G. Ing. Informática y del Software</p> <p>Programación Orientada a Objetos de G. Ing. Informática y del Software.</p> <p>Fundamentos de diseño del software de G. Ing. de Computadores</p> <p>Reconocidos dos tramos docentes</p>
Experiencia profesional	