GUÍA DOCENTE

PROGRAMACIÓN ORIENTADA A OBJETOS

DOBLE GRADO EN **MATEMÁTICA COMPUTACIONAL E INGENIERÍA DEL SOFTWARE**

MODALIDAD: PRESENCIAL

CURSO ACADÉMICO: 2023-2024

Denominación de la asignatura:	Programación Orientada a Objetos
Titulación:	DOBLE GRADO EN MATEMÁTICA COMPUTACIONAL E INGENIERÍA DEL SOFTWARE
Facultad o Centro:	Centro Universitario de Tecnología y Arte Digital
Materia:	Programación
Curso:	2º
Cuatrimestre:	1
Carácter:	OB
Créditos ECTS:	6
Modalidad/es de enseñanza:	Presencial
Idioma:	Castellano
Profesor/a - email	Miguel Angel Mesas Uzal / miguel.mesas@u-tad.com David Pinto Fernández / david.pinto@u-tad.com
Página Web:	http://www.u-tad.com/

DESCRIPCIÓN DE LA ASIGNATURA

Descripción de la materia

Esta asignatura pertenece a la materia de programación. Esta materia se dedica al estudio de las técnicas y los lenguajes de programación en los que se fundamentarán los estudios del grado de ingeniería del software.

Descripción de la asignatura

Esta asignatura le permite al alumno adquirir los conocimientos clave sobre el paradigma de programación orientado a objetos, realizar programas estructurados en clases y dotados de mecanismos como la herencia y el polimorfismo, así como trabajar con un lenguaje de programación orientado a objetos.

COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

Competencias (genéricas, específicas y transversales)

COMPTETENCIAS BÁSICAS Y GENERALES:

CG1 - Capacidad para entender, planificar y resolver problemas a través del desarrollo de soluciones informáticas

CG3 - Conocimiento de los fundamentos científicos aplicables a la resolución de problemas informáticos

CG4 - Capacidad para simplificar y optimizar los sistemas informáticos atendiendo a la comprensión de su complejida

CG9 - Capacidad para aprender, modificar y producir nuevas tecnologías informáticas

CG10 - Uso de técnicas creativas para la realización de proyectos informáticos

CB1 - 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

CB2 - 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

CB3 - 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

CB4 - Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado

CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores

con un alto grado de autonomía

COMPETENCIAS ESPECÍFICAS

CE1 - Conocimiento de la estructura de las computadoras, de los conceptos de codificación, manipulación, tratamiento de la información y uso de lenguajes de bajo nivel

CE7 - Conocimiento de los principales tipos de estructuras de datos y utilización de las librerías y de los técnicas algorítmicas asociadas a dichas estructuras junto con los órdenes de complejidad que caracterizan a dichas técnicas

CE8 - Conocimiento de los distintos paradigmas detrás de los lenguajes de programación

CE9 - Conocimiento de las estructuras de control, variables, sintaxis de programación y gestión del uso de la memoria de manera eficaz en el desarrollo de una aplicación informática

CE10 - Capacidad para manejar un gestor de versiones de código y generar la documentación de una aplicación de forma automática.

CE15 - Conocimiento de la tolerancia a los fallos, la adaptabilidad, el balance de carga y la predictividad del sistema para el desarrollo de aplicaciones distribuidas

CE17 - Conocimiento de las características de paralelización de tarjetas gráficas y de arquitecturas de altas prestaciones para el desarrollo de aplicaciones.

CE20 - Capacidad para testar el funcionamiento y funcionalidad de una aplicación informática, elaborando planes de pruebas y empleando técnicas de diseño y programación orientado a las pruebas

CE23 - Conocimiento de los principios de la inteligencia artificial y uso de algoritmos de búsqueda deterministas y máquinas de estado

Resultados de aprendizaje

Al acabar la titulación, el graduado o graduada será capaz de:

- Entender y manejar el concepto de memoria dinámica

- Identificar clases de objetos con los datos de un problema.

- Crear clases y objetos y manipularlos.

- Entender y utilizar los mecanismos de herencia, polimorfismo y sobrecarga de operadores.

- Identificar las relaciones entre clases en distintos casos de uso.

- Dominar un lenguaje de programación orientado a objetos.

- Dominar los patrones de programación

- Conocer las distintas formas de resolución de problemas desde el punto de vista de la algoritmia, como, por ejemplo, el esquema divide y vencerás,

- programación dinámica, backtracking o algoritmos genéticos.

- Estudiar la complejidad de un determinado algoritmo, interpretar dicha complejidad y analizar posibles optimizaciones.

- Codificar un programa que sea capaz de encontrar el camino óptimo que une dos nodos de un grafo siguiendo los distintos algoritmos de pathfinding.

- Crear y entrenar redes neuronales que solucionen problemas concretos.

CONTENIDO

Conceptos básicos de la programación orientada a objetos

Manejo de excepciones

TEMARIO

Tema 1: Introducción a la Programación Orientada a Objetos

Programación Procedural vs. Programación Orientada a Objetos.

Objetos y clases.

Propiedades de la orientación a objetos.

Tema 2: Programación en lenguaje Java

Entorno de desarrollo.o Operadores.

Bifurcaciones.o Bucles.o Variables.o Arrays.

Listas de argumentos.

Tema 3: Clases en Java

Concepto de clase.

Objetos y métodos.

Variables miembro.

Constructores.

Ámbito de las variables.

Uso de this.

Paquetes.

Relaciones de composición.

Tema 4: Herenciao Jerarquías de herencia.

Conversión de objetos.

Polimorfismo.

Clases abstractas.o Interfaces.

Tema 5: Manejo de excepciones

Excepciones estándar en Java.

Lanzamiento y captura de excepciones.

Creación de excepciones.

Tema 6: Programación de GUIs

Ventanas.

Botones.

Manejo de eventos.

Componentes gráficos de texto.

Gestores de disposición.

Tema 7: Entrada y salida de datos en Java

Clases para lectura y escritura.

Entrada y salida estándar.

La clase Scanner.

Ficheros de texto.

Ficheros binarios.

Tema 8: Plantillas y programación genérica.

Clases genéricas.

Métodos genéricos

ACTIVIDADES FORMATIVAS Y METODOLOGÍAS DOCENTES

Actividades formativas

Actividad Formativa	Horas totales	Horas presenciales
Clases teóricas / Expositivas	35,64	35,64
Clases Prácticas	18,91	18,91
Tutorías	4,00	2
Estudio independiente y trabajo autónomo del alumno	51,82	0,00
Elaboración de trabajos (en grupo o individuales)	33,82	0,00
Actividades de Evaluación	5,82	5,82
TOTAL	150,01	62,37

Metodologías docentes

Método expositivo o lección magistral

Aprendizaje de casos

Aprendizaje basado en la resolución de problemas

Aprendizaje cooperativo o colaborativo

Aprendizaje por indagación

Metodología Flipped classroom o aula invertida

Gamificación

Just in time Teaching (JITT) o aula a tiempo

Método expositivo o lección magistral

Método del caso

Aprendizaje basado en la resolución de problemas

Aprendizaje cooperativo o colaborativo

Aprendizaje por indagación

Metodología flipped classroom o aula invertida

Gamificación

DESARROLLO TEMPORAL

UNIDADES DIDÁCTICAS / TEMAS PERÍODO TEMPORAL

Tema 1: Introducción a la ProgramaciónOrientada a Objetos1 semana

Tema 2: Programación en lenguaje Java 2 semanas

Tema 3: Clases en Java 3 semanas

Tema 4: Herencia 3 semanas

Tema 5: Manejo de excepciones 1 semana

Tema 6: Programación de GUIs 2 semana

Tema 7: Entrada y salida de datos en Java 2 semana

Tema 8: Plantillas y programación genérica. 1 semana

SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN MÍNIMA RESPECTO A LA CALIFICACIÓN FINAL (%)	VALORACIÓN MÁXIMA RESPECTO A LA CALIFICACIÓN FINAL (%)
Evaluación de la participación en clase, en prácticas o en proyectos de la asignatura	0	30
Evaluación de trabajos, proyectos, informes, memorias	30	80
Prueba Objetiva	10	60

CRITERIOS ESPECÍFICOS DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	CONVOCATORIA ORDINARIA	CONVOCATORIA EXTRAORDINARIA

Evaluación de la participación en clase, en prácticas o en proyectos de la asignatura	0	0
Evaluación de trabajos, proyectos, informes, memorias	30	30
Prueba Objetiva	70	70

Consideraciones generales acerca de la evaluación

A lo largo del curso se plantea un proyecto que deberá ser entregado en la fecha indicada a través de la plataforma virtual. Este trabajo se evaluará con una demostración y supondrá un 30% de la nota. No se admiten entregas fuera de plazo,y si se aceptasen por causa justificada, el retraso supondrá una reducción considerableen la nota.

• Se plantearán prácticas que suponen un máximo de 10% de la nota final.

• La nota final en evaluación continua es, por tanto, la siguiente suma:0.30*proyecto+0.10*prácticas+0.6*prueba objetiva.

• Para aprobar la asignatura en la convocatoria ordinaria, es imprescindible que la nota final sea al menos 5.0 (sobre 10), es necesario una nota mínima de 4 en proyecto o prueba objetiva para ser compensable.

• Convocatoria ordinaria: Aquellos alumnos que no superen la nota 5,0 en el criterio anterior. Habrá un examen para la parte teórica que supondrá un 70% de la nota final. Si el alumno no superó la nota mínima de 4.0 en proyecto, tendrá oportunidad devolver a presentarlo.

• Convocatoria extraordinaria o única: Alumnos suspendidos en convocatoria ordinaria o alumnos a los que se les ha concedido la convocatoria única. Habrá un examen para la parte teórica que supondrá un 70% de la nota final. Si el alumno no superó la nota mínima de 4.0 en proyecto o no la realizó, tendrá oportunidad de presentarlo.

• No se conservarán calificaciones de ningún tipo entre distintos cursos académicos.

• No está permitido el uso de teléfonos móviles en el aula durante el período de evaluación continua, excepto indicación expresa en sentido contrario del profesor. Los ordenadores portátiles podrán utilizarse únicamente para actividades relacionadas con la asignatura. El profesor podrá retirar el derecho al uso del ordenador a aquellos alumnos que lo utilicen para actividades que no estén relacionadas con la asignatura (consulta de correos, noticias o redes sociales, consultao elaboración de actividades de otras asignaturas, etc.).

• Se demandará del alumno una participación activa, necesaria para el desarrollo de las clases.

• Se exigirá al alumno un buen comportamiento en todo momento durante el desarrollode las clases. El mal comportamiento que impida el normal desarrollo de la clase puede conllevar la expulsión del aula por un tiempo a determinar por el profesor

A lo largo del curso se plantea un proyecto que deberá ser entregado en la fecha indicada a través de la plataforma virtual. Este trabajo se evaluará con una demostración y supondrá un 30% de la nota. No se

admiten entregas fuera de plazo,y si se aceptasen por causa justificada, el retraso supondrá una reducción considerableen la nota.

• Se plantearán prácticas que suponen un máximo de 10% de la nota final.

• La nota final en evaluación continua es, por tanto, la siguiente suma:0.30*proyecto+0.10*prácticas+0.6*prueba objetiva.

• Para aprobar la asignatura en la convocatoria ordinaria, es imprescindible que la nota final sea al menos 5.0 (sobre 10), es necesario una nota mínima de 4 en proyecto o prueba objetiva para ser compensable.

• Convocatoria ordinaria: Aquellos alumnos que no superen la nota 5,0 en el criterio anterior. Habrá un examen para la parte teórica que supondrá un 70% de la nota final. Si el alumno no superó la nota mínima de 4.0 en proyecto, tendrá oportunidad devolver a presentarlo.

• Convocatoria extraordinaria o única: Alumnos suspendidos en convocatoria ordinaria o alumnos a los que se les ha concedido la convocatoria única. Habrá un examen para la parte teórica que supondrá un 70% de la nota final. Si el alumno no superó la nota mínima de 4.0 en proyecto o no la realizó, tendrá oportunidad de presentarlo.

• No se conservarán calificaciones de ningún tipo entre distintos cursos académicos.

• No está permitido el uso de teléfonos móviles en el aula durante el período de evaluación continua, excepto indicación expresa en sentido contrario del profesor. Los ordenadores portátiles podrán utilizarse únicamente para actividades relacionadas con la asignatura. El profesor podrá retirar el derecho al uso del ordenador a aquellos alumnos que lo utilicen para actividades que no estén relacionadas con la asignatura (consulta de correos, noticias o redes sociales, consultao elaboración de actividades de otras asignaturas, etc.).

• Se demandará del alumno una participación activa, necesaria para el desarrollo de las clases.

• Se exigirá al alumno un buen comportamiento en todo momento durante el desarrollode las clases. El mal comportamiento que impida el normal desarrollo de la clase puede conllevar la expulsión del aula por un tiempo a determinar por el profesor

BIBLIOGRAFÍA / WEBGRAFÍA

Básica:

Deitel & Deitel. Java Cómo programar.

Javier García de Jalón et. Al., Aprenda Java como si estuviera en primero. Universidad de Navarra.

Roger Cadenhead, Java 8. Anaya Multimedia.

Recomendada:

Matt Weisfeld, The Object-Oriented Thought Process.

Addison Wesley.David Etheridge, Java: The Fundamentals of Objects and Classes – An Introductionto Java Programming. Ventus Publishing ApS.

C. Thomas Wu, Object-Oriented Programming with Java. Mc Graw Hill.

Kathy Sierra & Bert Bates, Head First Java. O'Reilly.

David J. Eck, Introduction to Programming Using Java. Hobart and William SmithColleges

MATERIALES, SOFTWARE Y HERRAMIENTAS NECESARIAS

Tipología del aula
Aula teórica

Equipo de proyección y pizarra

Materiales:
Ordenador personal con Windows o Linux

Software:
JDK de Java

Eclipse