



DATOS IDENTIFICATIVOS

Programación II

Asignatura	Programación II			
Código	O06G150V01205			
Titulación	Grado en Ingeniería Informática			
Descriptor	Creditos ECTS	Selección	Curso	Cuatrimestre
	6	OB	1	2c
Lengua	Castellano			
Impartición	Gallego			
Departamento				
Coordinador/a	Cuesta Morales, Pedro			
Profesorado	Cuesta Morales, Pedro González Rufino, María Encarnación López Fernández, Hugo			
Correo-e	pcuesta@uvigo.es			
Web	http://fatic.uvigo.es			
Descripción general	Fundamentos básicos de programación orientada a objetos en JAVA. En esta asignatura no se utiliza el inglés como lengua de impartición ni en el material docente.			

Competencias

Código	
A2	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.
B8	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.
C4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
C5	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
C7	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
C12	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
C13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema
C14	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados
C25	Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software
C28	Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales
D1	I1: Capacidad de análisis, síntesis y evaluación
D2	I2: Capacidad de organización y planificación
D3	I3: Comunicación oral y escrita en la lengua nativa
D5	I5: Capacidad de abstracción: capacidad de crear y utilizar modelos que reflejen situaciones reales
D7	I7: Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos
D8	I8: Resolución de problemas
D9	I9: Capacidad de tomar decisiones
D10	I10: Capacidad para argumentar y justificar lógicamente las decisiones tomadas y las opiniones
D11	P1: Capacidad de actuar autónomamente
D12	P2: Capacidad de trabajar en situaciones de falta de información y/o bajo presión

D15	P5: Capacidad de relación interpersonal
D16	S1: Razonamiento crítico
D18	S3: Aprendizaje autónomo
D19	S4: Adaptación a nuevas situaciones
D20	S5: Creatividad
D22	S7: Tener iniciativa y ser resolutivo
D24	S9: Tener motivación por la calidad y la mejora continua

Resultados de aprendizaje

Resultados previstos en la materia	Resultados de Formación y Aprendizaje			
RA1. Conocer ampliamente el lenguaje de programación orientado a objetos de mayor utilidad para la industria en la actualidad.	A2	B8	C4	D1
			C5	D2
			C12	D5
			C13	D7
			C14	D8
				D9
				D11
				D12
				D16
				D18
				D19
				D20
				D22
	D24			
RA2. Conocer ampliamente el proceso de desarrollo asociado a un proyecto de complejidad básica realizado mediante programación orientada a objetos.	A2	B8	C7	D1
			C12	D2
			C13	D3
			C14	D5
			C25	D7
			C28	D8
				D9
				D10
				D11
				D12
				D15
				D16
				D18
	D19			
	D20			
	D22			
	D24			
RA3. Desarrollar software de calidad aplicando los fundamentos del paradigma de orientación a objetos.	A2		C14	D1
			C25	D2
			C28	D5
				D7
				D8
				D9
				D11
				D12
		D16		
		D18		
		D19		
		D20		

RA4. Dominar la comunicación dentro del grupo de trabajo, y la capacidad de iniciativa y de toma de decisiones en el trabajo realizado.	A2	B8	C14	D1
				D2
				D3
				D5
				D7
				D8
				D9
				D10
				D11
				D12
				D15
				D16
				D18
				D19
				D20
				D22
				D24

Contenidos

Tema	
Introducción al desarrollo orientado a objetos.	Clases y objetos. Encapsulación. Excepciones. Entrada/salida
Paradigma de desarrollo orientado a objetos.	Composición y herencia. Polimorfismo.
Archivos y pruebas	Genericidad. XML.

Planificación

	Horas en clase	Horas fuera de clase	Horas totales
Lección magistral	13.5	27	40.5
Resolución de problemas	27.5	27.5	55
Práctica de laboratorio	1.5	11	12.5
Resolución de problemas y/o ejercicios	10.5	31.5	42

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías

	Descripción
Lección magistral	Presencial: presentación, mediante medios audiovisuales, de los contenidos teóricos de cada tema. Este método se combinará con ejemplos ilustrativos de código y con la realización de preguntas para motivar e incrementar el interés del alumno. No presencial: revisión, comprensión y afianzamiento de los contenidos.
Resolución de problemas	El objetivo es que el alumno aplique los contenidos teóricos en la solución de problemas simples de programación. Presencial: resolución de pequeños problemas de programación. No presencial: resolución de pequeños problemas de programación.

Atención personalizada

Metodologías	Descripción
Resolución de problemas	Tiempo reservado para atender, guiar y resolver las dudas del alumnado.
Lección magistral	Tiempo reservado para atender, guiar y resolver las dudas del alumnado.

Evaluación

Descripción	Calificación	Resultados de Formación y Aprendizaje

Práctica de laboratorio	Se realizará una prueba individual de programación, con la cual se pretende evaluar la capacidad del alumno ante el desarrollo de software de calidad. Resultados: RA1, RA2, RA3.	30	A2	B8	C4 C5 C7 C12 C13 C14 C25 C28	D1 D2 D3 D5 D7 D8 D9 D10 D11 D12 D16 D18 D19 D20 D22 D24
Resolución de problemas y/o ejercicios	Cada tema tendrá una prueba individual, con la que se pretende comprobar si el alumno ha alcanzado los objetivos de dicho tema. Existirán tres pruebas, una por tema (bloque), valiendo cada prueba un 15% de la nota final (en total 45%). Resultados: RA1. Se realizarán tareas continuas en grupo (o individual) para la resolución de pequeños problemas de programación correspondientes a todos los temas de contenidos de la materia. Esta nota será proporcionada por el profesor de forma subjetiva (en total 25%). Resultados: RA2, RA3, RA4.	70	A2	B8	C4 C5 C7 C12 C13 C14 C25 C28	D1 D2 D3 D5 D7 D8 D9 D10 D11 D12 D15 D16 D18 D19 D20 D22 D24

Otros comentarios sobre la Evaluación

CRITERIOS DE EVALUACIÓN PARA ASISTENTES 1ª EDICIÓN DE ACTAS

- Todos los estudiantes que se presenten a cualquiera de las pruebas se entiende que siguen la asignatura de forma presencial y por lo tanto deberán de seguir el procedimiento de evaluación descrito anteriormente.
- Si un estudiante no se presenta a alguna de las pruebas se les asignará una calificación de 0 en ellas.

CRITERIOS DE EVALUACIÓN PARA NO ASISTENTES

- **Prueba:** Examen de preguntas objetivas
- **Descripción:** Cada tema tendrá una prueba individual, con la que se pretende comprobar si el alumno ha alcanzado los objetivos de dicho tema. Existirán tres pruebas, una por tema, valiendo cada prueba un 15% de la nota final.
- **% Calificación:** 45 %
- **Competencias Evaluadas:** CB2, CG8, CE4, CE5, CE12, CE13, CE14, CT1, CT2, CT5, CT7, CT8, CT9, CT11, CT12, CT16, CT18, CT19, CT20, CT22, CT24
- **Resultados de Aprendizaje evaluados:** RA1
- **Prueba:** Práctica de laboratorio.
- **Descripción:** Se realizará una prueba individual de programación, con la cual se pretende evaluar la capacidad del alumno ante el desarrollo de software de calidad.
- **% Calificación:** 55 %
- **Competencias Evaluadas:** CB2, CG8, CE4, CE5, CE7, CE12, CE13, CE14, CE25, CE28, CT1, CT2, CT3, CT5, CT7, CT8, CT9, CT10, CT11, CT12, CT16, CT18, CT19, CT20, CT22, CT24
- **Resultados de Aprendizaje evaluados:** RA1, RA2, RA3.

CRITERIOS DE EVALUACIÓN PARA 2ª EDICIÓN DE ACTAS Y FIN DE CARRERA

- Se aplicará el mismo sistema de evaluación para no asistentes

PROCESO DE CALIFICACIÓN DE ACTAS

- Para aplicar los porcentajes especificados es necesario obtener en cualquier prueba una nota igual o superior a 4 (sobre 10).
- Independientemente de la convocatoria, en caso de no superar alguna parte de la evaluación pero la puntuación global fuese superior a 5 (sobre 10), la calificación en actas será 4.

FECHAS DE EVALUACIÓN

- Las fechas y horarios de las pruebas de evaluación de las diferentes convocatorias son las especificadas en el calendario de pruebas de evaluación aprobado por la Junta de Centro para el curso 2019/2020.

OBSERVACIONES

- Se recuerda a todo el alumnado la prohibición del uso de dispositivos móviles en ejercicios y prácticas, en cumplimiento del artículo 13.2.d) del Estatuto del Estudiante Universitario, relativo a los deberes del estudiantado universitario, que establece el deber de "Abstenerse de la utilización o cooperación en procedimientos fraudulentos en las pruebas de evaluación, en los trabajos que se realicen o en documentos oficiales de la universidad."

Fuentes de información

Bibliografía Básica

Eckel, B., **Piensa en Java**, 4ª, Prentice Hall, 2007

Bibliografía Complementaria

Deitel, P. and Deitel, H., **Cómo programar en Java**, 10ª, Pearson, 2013

Jiménez Marín, A. y Pérez Montes, F.M., **Aprende a programar con JAVA**, 2ª, Paraninfo, 2016

Schildt, H., **Java 8**, 1ª, Anaya Multimedia, 2015

Recomendaciones

Asignaturas que se recomienda cursar simultáneamente

Informática: Algoritmos y estructuras de datos I/O06G150V01201

Asignaturas que se recomienda haber cursado previamente

Informática: Programación I/O06G150V01104