



Guía docente				
Datos Identificativos				2022/23
Asignatura (*)	Programación II		Código	614G03007
Titulación	Grao en Intelixencia Artificial			
Descriptor				
Ciclo	Periodo	Curso	Tipo	Créditos
Grado	2º cuatrimestre	Primero	Formación básica	6
Idioma	Castellano			
Modalidad docente	Presencial			
Prerrequisitos				
Departamento	Ciencias da Computación e Tecnoloxías da Información			
Coordinador/a	Cabrero Canosa, Mariano Javier	Correo electrónico	mariano.cabrero@udc.es	
Profesorado	, Alvarez Estevez, Diego Cabrero Canosa, Mariano Javier López Varela, Emilio Paz López, Alejandro Vilares Calvo, David	Correo electrónico	inigo.lopezriboo.botana@udc.es diego.alvareze@udc.es mariano.cabrero@udc.es e.lopezv@udc.es alejandro.paz.lopez@udc.es david.vilares@udc.es	
Web	campusvirtual.udc.es			
Descripción general	En esta asignatura el alumnado desarrollará las destrezas necesarias para decidir con criterio la combinación de estructuras de datos (lineales o árboles) y algoritmos más convenientes para resolver un determinado problema de forma eficiente en términos de recursos espaciales y temporales. Además, la asignatura introduce el paradigma de orientación a objetos, sus elementos y propiedades básicas como clases, objetos, herencia, polimorfismo, etc. y el paradigma de programación funcional, sus estructuras de datos características, y ámbitos de aplicación, enfatizando aquellos donde se logran planteamientos y resoluciones más fáciles que con otros paradigmas de programación			

Competencias del título	
Código	Competencias del título
A2	Capacidad para resolver problemas de inteligencia artificial que precisen algoritmos, aplicando correctamente metodologías de desarrollo software y diseño centrado en usuario/a.
A3	Capacidad para comprender y dominar los conceptos básicos de lógica, gramáticas y lenguajes formales para analizar y mejorar las soluciones basadas en inteligencia artificial.
B2	Que el alumnado sepa aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posea 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.
B4	Que el alumnado pueda transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
B5	Que el alumnado haya desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
B6	Capacidad para concebir, redactar, organizar, planificar, y desarrollar modelos, aplicaciones y servicios en el ámbito de la inteligencia artificial, identificando objetivos, prioridades, plazos recursos y riesgos, y controlando los procesos establecidos.
B7	Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
B8	Capacidad para diseñar y crear modelos y soluciones de calidad basadas en Inteligencia Artificial que sean eficientes, robustas, transparentes y responsables.
B9	Capacidad para seleccionar y justificar los métodos y técnicas adecuadas para resolver un problema concreto, o para desarrollar y proponer nuevos métodos basados en inteligencia artificial.
C2	Capacidad de trabajo en equipo, en entornos interdisciplinares y gestionando conflictos.
C3	Capacidad para crear nuevos modelos y soluciones de forma autónoma y creativa, adaptándose a nuevas situaciones. Iniciativa y espíritu emprendedor.

Resultados de aprendizaje



Resultados de aprendizaje	Competencias del título		
Llevar a cabo el proceso que permite, desde la abstracción, implementar código de alta calidad.	A2	B2 B5 B7	C3
Aplicar programación modular para resolver problemas específicos en el ámbito de IA.	A2	B6 B8 B9	C3
Adquirir competencias para resolver problemas de forma metodológica y práctica.		B5 B7	C3
Identificar y tener la capacidad para seleccionar en un entorno práctico las principales librerías en el campo de IA y Ciencia de Datos.		B2 B5 B9	
Comprender los conceptos básicos de la programación funcional y el lambda cálculo y demostrar en qué medida su aplicación favorece el desarrollo de aplicaciones en IA.	A3	B8 B9	
Adquirir las competencias para analizar la complejidad computacional de un determinado algoritmo, así como desarrollar las capacidades necesarias para escoger la combinación de estructuras de datos y estrategia de resolución más apropiada para resolver de modo eficiente (en términos de recursos espaciales y temporales) un determinado problema.	A2	B4 B6 B7	C2 C3
Analizar las alternativas para afrontar e identificar que aspectos pueden abordarse con IA y cuales no.	A2 A3	B6 B8 B9	C2
Comprender los principios necesarios para construir soluciones completas, escalables y robustas, centradas en el/la usuario/a, en las que los componentes de IA encajan como parte de un todo.	A2	B2 B4 B6 B8 B9	C3
Manejar técnicas y herramientas de prueba para asegurar la calidad de los resultados.		B8 B9	C2

Contenidos	
Tema	Subtema
Introducción a la orientación a objetos	Clases y objetos. Métodos. Herencia. Interfaces y Polimorfismo.
Técnicas de diseño de programas	Abstracción y especificación Módulos Excepciones y eventos
Tipos abstractos de datos lineales	Listas Pilas Colas y Colas de Prioridad
Tipos abstractos de datos no lineales	Árboles Árboles Binarios de búsqueda Árboles AVL
Introducción a la programación funcional	Funciones puras Recursión Inmutabilidad

Planificación



Metodoloxías / probas	Competencias	Horas presenciais	Horas no presenciais / traballo autónomo	Horas totais
Prácticas de laboratorio	A2 A3 B2 B6 B7 B8 B9 C2 C3	20	40	60
Solución de problemas	A2 A3 B6 B7	10	17	27
Prueba objetiva	B2 B4 B5 B7 B9	2	0	2
Sesión magistral	A2 A3 B5 B9	30	30	60
Atención personalizada		1	0	1
(*) Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos				

Metodoloxías	
Metodoloxías	Descrición
Prácticas de laboratorio	Realización y entrega de distintas prácticas de programación para fomentar o traballo continuado. Proporcionarase o enunciado e as especificacións que deberán respectarse estritamente. Durante as clases o profesor supervisará o traballo solucionando dúbidas e corrixiendo erros de interpretación, malos hábitos de programación e erros de sintaxe, etc.
Solución de problemas	Presentación y resolución de supuestos prácticos sencillos para afianzar los conceptos teóricos. Podrán requerir el desarrollo de código. Se fomentará la participación de los alumnos poniendo en común las distintas soluciones para promover el diálogo aberto y la valoración de soluciones. Además se podrán formular ejercicios adicionales que el estudiante deberá resolver y comentar/corregir con el profesor.
Prueba objetiva	Evaluación sumativa del alumno mediante un examen final al término del cuatrimestre. Será eminentemente práctico para que el alumno pueda demostrar que adquirió los conocimientos necesarios de diseño de programas, orientación a objetos y utilización de estructuras de datos, y que entrenó lo suficiente con ellos como para poseer las habilidades precisas para resolver supuestos prácticos que implique la aplicación de los dichos conocimientos.
Sesión magistral	Exposición oral complementada con el uso de medios audiovisuales y la introducción de algunas preguntas dirigidas a los estudiantes, con la finalidad de transmitir conocimientos y facilitar el aprendizaje. Constarán de una breve descripción de los contenidos temáticos y de los objetivos básicos perseguidos, e interrelaciones con otros conceptos previamente adquiridos para pasar a desarrollar los contenidos teóricos.

Atención personalizada	
Metodoloxías	Descrición
Solución de problemas Prácticas de laboratorio	El desarrollo de las sesiones de solución de problemas y los laboratorios de prácticas se realizará atendiendo al progreso de los alumnos en las capacidades de comprensión y asimilación de los contenidos impartidos. El avance general de la clase se compaginará con una atención específica a aquellos alumnos que presenten mayores dificultades en la tarea del aprendizaje y con un apoyo adicional a aquellos otros que presenten mayor desparpajo y deseen ampliar conocimientos. En el que respeta a las tutorías individuales, dado su carácter personalizado deben dedicarse a aclarar los conceptos ya expuestos y serán una vía de interacción que permitirá extraer conclusiones respecto al grado de asimilación de la materia. Se realizarán de manera presencial. Excepcionalmente se realizará a través de las herramientas corporativas de la UDC (email, Teams, etc.) respetando los horarios docentes.

Evaluación			
Metodoloxías	Competencias	Descrición	Calificación
Prueba objetiva	B2 B4 B5 B7 B9	Realización obligatoria. Necesario aprobar la prueba para superar a materia.	60
Prácticas de laboratorio	A2 A3 B2 B6 B7 B8 B9 C2 C3	Realización y calificación según las condiciones establecidas en el enunciado de cada práctica. Necesario aprobar las practicas para superar la materia.	40



Observaciones evaluación

Trabajos prácticos y solución de problemas

De acuerdo al artículo 14, apartado 4, de la normativa*, el plagio de los trabajos llevará una nota global de NO APTO (0), tanto al estudiante que presente material copiado como a quién lo facilitara, y la cualificación de SUSPENSO en la oportunidad. Si las prácticas u otras actividades se realizan en grupo, todos los miembros del grupo responderán de forma solidaria del trabajo realizado y entregado y de sus posibles consecuencias. Matrícula a tiempo parcial

Los alumnos matriculados a tiempo parcial tendrán que entregar las actividades evaluables en las condiciones y plazos específicos que se establecerán. Será deber del estudiante comunicar su situación al profesorado. No presentado

Quien no concurra a la prueba objetiva en el período oficial de evaluación tendrá la condición de ?No presentado? (NP). * Normativa de evaluación, revisión y reclamación de las cualificaciones de los estudios de grado y máster universitario, aprobada por el Consejo de Gobierno de la Universidade da Coruña el 19 de diciembre de 2013.

Fuentes de información

Básica	- Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser (2013). Data Structures and Algorithms in Python. John Wiley & Sons - Kenneth A. Lambert (2013). Fundamentals of Python: Data Structures. Course Technologies
Complementaria	- Bradley N. Miller, David L. Ranum. (2013). Problem Solving with Algorithms and Data Structures using Python.. Franklin, Beedle & Associates - Benjamin Baka (2017). Python Data Structures and Algorithms: Improve application performance with graphs, stacks, and queues.. Packt Publishing - Camilo Chacon (2021). Computación y Programación Funcional. Marcombo Sitio web da contorna de desenvolvemento Spyder: https://www.spyder-ide.org/ Sitio web da contorna de desenvolvemento Spyder: https://www.spyder-ide.org/

Recomendaciones

Asignaturas que se recomienda haber cursado previamente

Programación I/614G03006

Matemática Discreta/614G03003

Asignaturas que se recomienda cursar simultáneamente

Asignaturas que continúan el temario

Otros comentarios

(*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías