UNIVERSIDADE DA CORUÑA

Guia docente				
Datos Identificativos				2021/22
Asignatura (*)	Algoritmos		Código	614G01011
Titulación	Grao en Enxeñaría Informática			
Descriptores				
Ciclo	Periodo	Curso	Tipo	Créditos
Grado	1º cuatrimestre	Segundo	Obligatoria	6
Idioma	CastellanoInglés			
Modalidad docente	Híbrida			
Prerrequisitos				
Departamento	Ciencias da Computación e Tecnoloxías da InformaciónComputación			
Coordinador/a	Valderruten Vidal, Alberto		Correo electrónico	alberto.valderruten@udc.es
Profesorado	Aguado Martin, Maria Felicidad Cabalar Fernandez, Jose Pedro Cancela Barizo, Brais Casanova Crespo, Jose Maria Gómez Rodríguez, Carlos Hernandez Pereira, Elena Maria Jorge Castro, Jose Santiago Perez Vega, Gilberto Valderruten Vidal, Alberto Vidal Martin, Concepcion		Correo electrónico	felicidad.aguado@udc.es pedro.cabalar@udc.es brais.cancela@udc.es jose.casanova.crespo@udc.es carlos.gomez@udc.es elena.hernandez@udc.es santiago.jorge@udc.es gilberto.pvega@udc.es alberto.valderruten@udc.es concepcion.vidalm@udc.es
Web	moodle.udc.es/course/view.php?id=55374			
Descripción general	La asignatura de Algoritmos permite al estudiante de Ingeniería informática profundizar en el uso de técnicas de diseño de algoritmos teniendo en cuenta factores cualitativos y cuantitativos en la evaluación de los mismos. Por una parte completa la formación en la elaboración de programas eficientes y correctamente estructurados, y por otra parte permite abordar las técnicas de diseño más utilizadas en la resolución de los problemas que puede encontrar el ingeniero. Es de destacar que la realización de experimentos de medición de tiempos de ejecución sobre los distintos algoritmos analizados aporta un enfoque empírico que suele ser muy valorado por el estudiante, que puede así constatar la interpretación concreta de las complejidades encontradas. Las dificultades planteadas por algunos casos estudiados permiten una reflexión complementaria sobre aspectos como la gestión de recursos informáticos, detalles de ejecución de procesos, arquitecturas y sistemas operativos utilizados, etc. También es destacable el estudio y análisis de un conjunto importante de algoritmos fundamentales, cubriendo un amplio espectro de técnicas algorítmicas y de sus aplicaciones. La posibilidad de usar distintas técnicas en la resolución de algunos problemas lleva naturalmente a pensar en ventajas e inconvenientes de las distintas estrategias, y en la necesidad de saber elegir la que mejor se adapte a cada situación. Por último es importante profundizar en el rigor necesario para desarrollar no sólo soluciones que se adapten a unas especificaciones dadas, sino además que lo hagan de modo eficiente desde el punto de vista de los recursos informáticos necesarios. Resulta fundamental la ilustración mediante varios casos prácticos en los que la existencia de algoritmos eficientes conocidos lleva a desechar los diseños alternativos por muy naturales que puedan resultar a primera vista.			

Plan de contingencia	1. Modificaciones en los contenidos Sin cambios. 2. Metodologías *Metodologías docentes que se mantienen Trabajos tutelados *Metodologías docentes que se modifican Sesión magistral: Se realizará de forma no presencial, a través de Teams y/o de vídeos grabados. Prueba de respuesta breve: Se realizará de forma no presencial, a través de Teams y Moodle. Prácticas de laboratorio: Se realizarán de forma no presencial, usando Teams. Se recomienda una configuración del portátil con linux, gcc y svn. Solución de problemas: Se realizará de forma no presencial, usando Teams y Moodle. Prueba objetiva: Se realizará de forma no presencial, usando Teams e Moodle, además de la configuración recomendada del portátil con linux, gcc e svn para la parte práctica. 3. Mecanismos de atención personalizada al alumnado Correo electrónico: diariamente, para consultas. Moodle: diariamente, para acceder a los materiales de clase, consultar el calendario o hacer uso de los foros. Teams: durante las horas de teoría, TGR o práctica previstas en el horario de la asignatura; tutorías en grupo sobre la teoría (2h por semana) y sobre las prácticas (2h por semana); en lo referente a las tutorías individuales, fuera de las horas de clase la atención se mantiene en los horarios de tutorías previa solicitud a través del correo electrónico. 4. Modificaciones en la evaluación Todas las pruebas (evaluación continua y prueba objetiva) son en modalidad non presencial. *Observaciones de evaluación: Se mantienen las previsiones de la guía docente. Se ofrecerá una fecha alternativa a los estudiantes que tengan problemas logísticos en el momento de las pruebas. 5. Modificaciones de la bibliografía o webgrafía Sin cambios

Competencias / Resultados del título	
Código	Competencias / Resultados del título
A12	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.
A13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
B3	Capacidad de análisis y síntesis
C3	Utilizar las herramientas básicas de las tecnologías de la información y las comunicaciones (TIC) necesarias para el ejercicio de su profesión y para el aprendizaje a lo largo de su vida.

Resultados de aprendizaje	
Resultados de aprendizaje	Competencias / Resultados del título

Saber aplicar las técnicas de análisis de la complejidad de los algoritmos.	A12 A13	B3	
Reconocer la importancia del estudio de la complejidad de los algoritmos y saber realizar estudios empíricos para determinarla.	A12 A13	B3	C3
Conocer las técnicas más utilizadas en el diseño de los algoritmos.	A12	B3	
Comprender elementos de estudio sobre la complejidad computacional.	A12 A13	B3	
Utilizar diferentes modelos de computación y niveles de abstracción necesarios para el diseño de algoritmos.	A12	B3	
Identificar estructuras de datos adaptadas a los algoritmos estudiados para obtener implementaciones más eficientes y robustas.	A13	B3	C3

Contenidos	
Tema	**Subtema**
Tema 1 Título del tema: Análisis de Algoritmos. Código: T1 Presentación: En este primer tema se plantea el análisis de la complejidad de los algoritmos como uno de los principales objetivos del curso. En síntesis, se trata de añadir a los criterios que ya deben resultar familiares, los de estructuración y de corrección de los programas, el de la efi ciencia de los algoritmos.	Unidades de contenido: 1. Análisis de la eficiencia de los algoritmos: Notaciones asintóticas, Modelo de computación, Verificación empírica del análisis. 2. Cálculo de los tiempos de ejecución: Análisis de los casos peor y medio, Cálculo de la O, Resolución de recurrencias.
Tema 2 Título del tema: Estructuras de datos. Código: T2 Presentación: En este segundo tema se propone una revisión de las estructuras de datos básicas (pilas, listas, colas, árboles, conjuntos y grafos) con el objetivo de estudiar todas las implicaciones que conlleva su uso en cuanto a las complejidades espacial y temporal. Igualmente se profundiza en el estudio de estructuras interesantes desde el punto de vista del tiempo de ejecución: las tablas de dispersión y los montículos, estructura ésta última a la que recurriremos más adelante cuando se trate de implementar mejoras en algoritmos de grafos y algún caso de programación dinámica. La complejidad de la operación de búsqueda puede servir como hilo conductor en buena parte de este tema. Conviene en una introducción de esta parte del curso el insistir en los criterios de estructuración que debemos mantener en el diseño de cualquier aplicación, motivando el uso de tipos de datos abstractos y su consiguiente implementación mediante módulos. El objetivo es dar así las líneas generales de lo que se considera la disciplina de programación que debe exigirse al estudiante para la realización de las prácticas.	Unidades de contenido: 1. Pilas, colas, listas. 2. Árboles, montículos. 3. Dispersión (hashing). 4. Conjuntos disjuntos. 5. Grafos (representación).

Tema 3 Título del tema: Algoritmos sobre secuencias y conjuntos de datos Código: T3 Presentación: El problema de la ordenación de una secuencia de elementos se convierte en esta parte del curso en una excusa ideal tanto para estudiar la complejidad de varios tipos de algoritmos como para presentar diferentes estrategias de diseño de algoritmos que se pueden extrapolar para la resolución de otros problemas. Uno de los algoritmos a los que se le dedicará especial atención es a la ordenación rápida, ya que permite introducir la característica fundamental de los algoritmos aleatorios, que se comportan de forma distinta al ejecutarse con una misma entrada. Una consecuencia directa es que el calificativo de peor caso o mejor caso para una entrada deja de tener sentido, aspecto que es importante debatir en clase.	Unidades de contenido: 1. Algoritmos de búsqueda. 2. Algoritmos de ordenación: Inserción, Shell, Montículos (heapsort), Fusión (mergesort), Ordenación Rápida (quicksort). 3. Algoritmos aleatorios.
Tema 4 Título del tema: Algoritmos voraces Código: T4 Presentación: En este tema se estudian algoritmos ávidos o voraces. Una vez explicada la técnica a través de sus características generales que presentaremos con la ayuda de algún ejemplo, se estudiarán los algoritmos más representativos de esta categoría: los algoritmos de grafos, una solución al problema de la mochila y algún problema de plani ficación de tareas.	Unidades de contenido: 1. Problema de la mochila. 2. Algoritmos de grafos: Ordenación Topológica, Árbol de recubrimiento mínimo, Caminos mínimos. 3. Problemas de plani ficación de sistemas informáticos.
Tema 5 Título del tema: Diseño de algoritmos por inducción Código: T5 Presentación: En este punto, ya se habrá visto a lo largo del curso varios algoritmos que siguen la estrategia divide y vencerás: ordenación por fusión y ordenación rápida, búsqueda dicotómica, suma de la subsecuencia máxima. . . El trabajo propuesto en la primera unidad de este tema consiste básicamente en generalizar los planteamientos de dicha estrategia identifi cando sus distintas características en cada uno de los algoritmos propuestos. En la segunda unidad del tema se plantea usar una estrategia ascendente mediante la búsqueda de una solución general a partir de las soluciones de subproblemas elementales. Desde el punto de vista de la efi ciencia se cuestionará el uso de técnicas descendentes como divide y vencerás en determinadas situaciones. Mientras que con la opción de la programación dinámica se buscará un compromiso que permita, cuando sea posible, una optimización de la cantidad de memoria requerida por el algoritmo.	Unidades de contenido: 1. Divide y Vencerás. 2. Programación dinámica: Principio de optimalidad, Problema de la mochila.

Tema 6 Título del tema: Exploración de grafos. Código: T6 Presentación: El objetivo de este tema es el de dar una visión más amplia de las aplicaciones de los grafos en el tratamiento de problemas de diversa índole, así como la de no dejar de lado las técnicas algorítmicas ligadas al desarrollo de importantes áreas de la computación como la inteligencia artificial. Los algoritmos de grafos vistos en el tema de algoritmos voraces (T4) coinciden en realizar un recorrido de todos los nodos del grafo. Se insistirá entonces en cómo mejorar los tiempos de ejecución de los algoritmos que se presenten evitando un análisis exhaustivo de todos los nodos.	Unidades de contenido: 1. Juegos de estrategia. 2. Recorridos. 3. Algoritmos con retroceso.
Tema 7 Título del tema: Complejidad Computacional. Código: T7 Presentación: En este último tema planteamos un razonamiento sobre el conjunto de los algoritmos capaces de resolver cada tipo de problema. Hablaremos de las complejidades de los problemas, de cotas inferiores para la complejidad de los problemas y de NP-compleción, en definitiva, de las principales técnicas y conceptos que se utilizan en el estudio de la complejidad computacional.	Unidades de contenido: 1. NP-Completitud, Problemas NP-completos

Planificación				
Metodologías / pruebas	Competencias / Resultados	Horas lectivas (presenciales y virtuales)	Horas trabajo autónomo	Horas totales
Sesión magistral	A12 A13 B3	28.75	28.75	57.5
Prueba de respuesta breve	A12 A13 B3	1.25	6.25	7.5
Prácticas de laboratorio	A12 A13 B3 C3	19	19	38
Trabajos tutelados	A12 A13 B3 C3	4	2	6
Solución de problemas	A12 A13 B3	5	10	15
Prueba objetiva	A12 A13 B3 C3	4	20	24
Atención personalizada		2	0	2
(*)Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos				

Metodologías	
Metodologías	Descripción
Sesión magistral	Clases magistrales en la exposición de los conocimientos teóricos utilizando diferentes recursos: la pizarra, transparencias, proyecciones, demostraciones y la facultad virtual. Puede incluir conferencia invitada.
Prueba de respuesta breve	En general consiste en la realización de ejercicios sobre la ejecución de casos sobre los algoritmos estudiados o sobre su adaptación a otras situaciones. Estas pruebas son evaluadas.
Prácticas de laboratorio	Prácticas diseñadas por el profesor basadas en los conocimientos que el estudiante va adquiriendo en las clases magistrales y que por tanto las complementan. Los estudiantes desarrollarán estos trabajos en parejas a lo largo del curso, e individualmente en una práctica final que se incluye en la prueba objetiva. Se implementarán programas que ilustren los problemas relacionados con el tema. Se pedirá el informe de resultados para su evaluación. A lo largo de las horas asignadas a cada práctica se evaluarán los informes de la práctica anterior.

Trabajos tutelados	Trabajos tutelados propuestos por el profesor y desarrollados por los estudiantes o bien en grupo o bien individualmente.
Solución de problemas	Se desarrollarán ejemplos sobre los conocimientos teóricos relacionados con el tema, y se resolverán dudas. Se evaluará individualmente la resolución de algunos problemas.
Prueba objetiva	Se evaluará el dominio de los conocimientos teóricos y operativos de la materia. Se evaluará igualmente la práctica individual final.

Atención personalizada	
Metodologías	**Descripción**
Trabajos tutelados Prácticas de laboratorio Solución de problemas	Clases de problemas en grupos reducidos: Se desarrollarán ejemplos sobre los conocimientos teóricos relacionados con el tema, y se resolverán dudas. Trabajos tutelados bien individuales bien en grupo sobre algún aspecto del tema a estudiar. Son controlados por parte del profesor mediante tutorías en grupo y controles de evaluación. Prácticas de aula de informática: Se implementarán programas que ilustren los problemas relacionados con el tema. Se pedirán el informe de resultados para su evaluación. En lo referente a las tutorías individuales, fuera de las horas de clase la atención se mantiene en los horarios de tutorías a través de los siguientes canales: - Correo electrónico, para consultas de respuesta breve. - Teams: encuentros virtuales preferentemente previa solicitud a través del correo electrónico.

Evaluación			
Metodologías	**Competencias / Resultados**	**Descripción**	**Calificación**
Prueba de respuesta breve	A12 A13 B3	2 pruebas escritas de evaluación continua, en las que se evaluará el dominio de los contenidos de los trabajos académicos a revisar. Se realizarán durante las horas de teoría que se incluyen en la planificación inicial de la asignatura presentada al inicio del curso.	10
Prueba objetiva	A12 A13 B3 C3	Se evaluará el dominio de los conocimientos teóricos y operativos de la materia. Examen individual de teoría: 50% Examen individual de prácticas: 20% Para ser convocada/o al examen de prácticas de la 1ª oportunidad, es necesaria la entrega en plazo de las prácticas de laboratorio.	70
Prácticas de laboratorio	A12 A13 B3 C3	4 prácticas realizadas en parejas, en las que se evaluará: estructuración de los programas, calidad de la documentación, claridad, adecuación y explicación de los resultados. La entrega en tiempo y forma de las prácticas es condición necesaria para acceder al examen individual final de prácticas de la 1ª oportunidad. La evaluación se realiza mediante tutorías de seguimiento de las prácticas, durante las horas de práctica.	10
Solución de problemas	A12 A13 B3	Evaluación de 2 ó 3 trabajos en los que se desarrollarán ejemplos sobre los conocimientos teóricos relacionados con el tema tras resolver dudas. Se realizarán durante las horas de Trabajo en Grupo Reducido (TGR) planificadas a lo largo del curso, pudiendo en algún caso completarse en horas no presenciales.	10

Otros			

Observaciones evaluación
En la 2ª oportunidad el estudiante podrá presentarse nuevamente tanto al examen de prácticas como al examen teórico (partes previstas en la prueba objetiva). El examen individual de prácticas (prueba objetiva) tendrá lugar en el mismo día fijado para el examen de la asignatura y podrán establecerse distintos turnos dependiendo del número de estudiantes matriculados; es necesario disponer, previamente, del material de todas las prácticas realizadas a lo largo del curso en la cuenta de usuario de cada estudiante. Presentarse al examen de prácticas o al examen de teoría supondrá una calificación distinta al "no presentado" en el acta correspondiente. Según lo previsto por la Norma que regula el régimen de dedicación al estudio de los estudiantes de Grado de la UDC, los estudiantes podrán optar por una dedicación a tiempo parcial. Su implantación en el ámbito de esta asignatura supondrá que, de forma general, la calificación que figurará en el acta será la mejor entre la obtenida según el criterio especificado en esta sección de la guía docente, y la obtenida únicamente con la prueba objetiva ponderando en un 70% el examen de teoría y en un 30% el examen de prácticas. En la oportunidad adelantada de diciembre el 100% de la evaluación corresponderá a un examen específico de teoría que incluirá cuestiones relacionadas con las prácticas.

Fuentes de información	
Básica	- M. A. Weiss (1995). Estructuras de Datos y Algoritmos. Addison Wesley - U. Manber (1989). Introduction to Algorithms - A Creative Approach. Addison Wesley - G. Brassard y P. Bratley (1997). Fundamentos de Algoritmia. Prentice Hall
Complementária	- F. Aguado, F. Gago, M. Ladra, G. Pérez, C. Vidal y A. M. Vieites (2018). Problemas resueltos de Combinatoria. Laboratorio con SageMath. Paraninfo - B. W. Kernighan y D. M. Ritchie (1991). El lenguaje de programación C, 2ª edición. Prentice Hall - T. H. Cormen, C. E. Leiserson y R. L. Rivest (1990). Introduction to Algorithms. MIT Press - R. Peña Marí (2005). Diseño de Programas. Formalismo y Abstracción. Tercera edición.. Pearson Prentice Hall - R. Sedgewick (1988). Algorithms. Addison Wesley

Recomendaciones
Asignaturas que se recomienda haber cursado previamente
Matemática Discreta/614G01004 Programación II/614G01006
Asignaturas que se recomienda cursar simultáneamente
Paradigmas de Programación/614G01014
Asignaturas que continúan el temario
Concurrencia y Paralelismo/614G01018 Sistemas Inteligentes/614G01020
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