UNIVERSIDADE DA CORUÑA

Guía Docente					
Datos Identificativos					2014/15
Asignatura (*)	Algoritmos			Código	614111206
Titulación					
Descriptores					
Ciclo	Período	Curso		Tipo	Créditos
1º e 2º Ciclo	1º cuadrimestre	Segundo		Obrigatoria	5
Idioma	Castelán				
Prerrequisitos					
Departamento	Computación				
Coordinación			Correo electrónico		
Profesorado			Correo electrónico		
Web	www.madsgroup.org/docencia/alg				
Descrición xeral	La asignatura de Algoritmos permite al estudiante de ingeniería informática profundizar en las técnicas de diseño de los 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 efi cientes 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 de los distintos programas 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 di ficultades planteadas por algunos casos estudiados permiten una refl exión 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 aplicar 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 adapta 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 e ficiente 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 e ficientes conocidos lleva a desechar los diseños alternativos por muy naturales que puedan resultar a primera vista.				

Competencias da titulación	
Código	Competencias da titulación

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Reconocer la importancia del estudio de la complejidad de los algoritmos.	A1 A3 A6	B2 B3 B9 B12	C6
Conocer los conceptos básicos relacionados con las técnicas de análisis de la complejidad de los algoritmos.	A1 A8	B2 B12	
Identi car estructuras de datos adaptadas a los algoritmos estudiados para obtener implementaciones más efi cientes.	A2 A8	B2	C6
Conocer las técnicas más utilizadas en el diseño de los algoritmos.	A1 A3 A6	B4 B11 B12 B15	C6

Diferenciar diferentes modelos de computación y niveles de abstracción necesarios para el diseño de algoritmos.	A1 A2 A3	B2 B3 B4	
Utilizar elementos de estudio sobre la complejidad computacional.	A1 A3	B1 B2 B4 B9	C6
Efectuar estudios empíricos para determinar la complejidad de un algoritmo.	A3	B2 B3 B5 B7 B11 B12	C6
Mostrar interés sobre el análisis de los algoritmos.		B9 B11	C6

Contidos	
Temas	**Subtemas**
Tema 1 Título del tema: Análisis de Algoritmos. Código: T1 Presentación: En este primer tema se plantea el análisis de la complejidad de los algo- ritmos como uno de los principales objetivos del curso. En de nitiva, se trata de añadir a los criterios que ya deben resultar familiares de estructuración y de corrección de los programas el de la efi ciencia de los algoritmos.	Unidades de contenido: 1. Análisis de la efi ciencia de los algoritmos: Notaciones asintóticas, Modelo de com- putación, Verifi cación 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 la ordenación rápida, ya que permite introducir la característica fundamental de los algoritmos aleatorios que se pueden comportar de forma distinta cuando se aplican dos veces a una misma entrada. Una consecuencia directa es que el cali ficativo 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 (merge-sort), 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. Algoritmos de grafos: Árbol de recubrimiento mínimo, Caminos mínimos. 2. Problema de la mochila. 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 habra 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 e 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 apli- caciones 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 arti cial. 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 de alguna manera el análisis exhaustivo de todos los nodos.	Unidades de contenido: 1. Juegos de estrategia 2. Recorridos 3. Algoritmos con retroceso.

Tema 7 Título del tema: Algoritmos paralelos. Código: T7 Presentación: El número creciente de ordenadores paralelos o sistemas distribuidos que pueden dedicarse a la realización concurrente de un proceso de computación justi ca la introducción al uso de estas técnicas que podemos hacer en el marco de este curso. Lo primero que hay que hacer es modi car el modelo de computación visto en el primer tema (T1), con el que veníamos trabajando, para introducir la noción de memoria compartida.	Unidades de contenido: 1. Modelos de computación paralela. 2. Algoritmos para sistemas de memoria compartida, Algoritmos para redes de inter-conexión.
Tema 8 Título del tema: Complejidad Computacional. Código: T8 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 de nitiva, de las principales ténicas y conceptos que se utilizan en el estudio de la complejidad computaciional.	Unidades de contenido: 1. NP-Completitud, Problemas NP-completos

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Proba de resposta breve	1.5	6	7.5
Traballos tutelados	2	12	14
Prácticas de laboratorio	12	12	24
Discusión dirixida	2	2	4
Proba obxectiva	2	12	14
Sesión maxistral	21	31.5	52.5
Solución de problemas	4.5	4.5	9
Atención personalizada	0		0
*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado			

Metodoloxías	
Metodoloxías	Descrición
Proba de resposta breve	En general consiste en la evaluación de trabajos tutelados.
Traballos tutelados	Trabajos tutelados propuestos por el profesor y desarrollados por los estudiantes o bien en grupo o bien individualmente. Estos trabajos tutelados podrán ser evaluados mediante controles tipo test o en tutorías personalizadas.
Prácticas de laboratorio	Prácticas diseñadas por el profesor basadas en los conocimientos que el estudiante va adquiriendo. Los estudiantes desarrollarán estos trabajos bien individualmente bien en grupo. Se implementarán programas que ilustren los problemas relacionados con el tema. Se pedirá el informe de resultados para su evaluación.
Discusión dirixida	Se desarrollarán ejemplos sobre los conocimientos teóricos relacionados con el tema, y se resolverán dudas.
Proba obxectiva	Se evaluará el dominio de los conocimientos teóricos y operativos de la materia.

Sesión maxistral	Clases magistrales en la exposición de los conocimientos teóricos usándose diferentes recursos: la pizarra, transparencias, proyecciones, apuntes y la facultad virtual.
Solución de problemas	Clases de problemas para asentar las enseñanzas. El profesor preparará los problemas que los estudiantes resolverán en su tiempo de trabajo y estudio personal. Posteriormente, a petición de los estudiantes o a criterio del profesor, se solucionarán durante las clases los más críticos.

Atención personalizada	
Metodoloxías	**Descrición**
Solución de problemas Traballos tutelados Prácticas de laboratorio Discusión dirixida	Clases de problemas en el aula: 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 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.

Avaliación		
Metodoloxías	**Descrición**	**Cualificación**
Proba de resposta breve		0
Prácticas de laboratorio	Examen individual de prácticas	35
Proba obxectiva	Se evaluará el dominio de los conocimientos teóricos y operativos de la materia.	65
Outros		

Observacións avaliación
Tanto en el examen de prácticas como en el teórico, se considerará insuficiente una calificación inferior a 4/10. En la convocatoria de junio, las prácticas de laboratorio se evaluarán durante tutorías de seguimiento que se realizarán en el horario de prácticas. Esta evaluación continua supondrá 3/7 partes de la evaluación de las prácticas. Las prácticas son obligatorias, excepto para los estudiantes repetidores que tengan una nota de prácticas obtenida en los dos últimos cursos. En las convocatorias de septiembre y diciembre, el examen teórico (prueba objetiva) supondrá un 65% de la evaluación, ya que no se realizan las pruebas escritas de evaluación continua (pruebas de respuesta breve). Presentarse a un examen de prácticas o a un examen de teoría implica presentarse a la convocatoria correspondiente. Información sobre el examen de prácticas en las convocatorias de diciembre y septiembre: para estas convocatorias, el examen de prácticas tendrá lugar en los laboratorios de la primera planta tras la finalización del examen de teoría (en el mismo día); es necesario tener, previamente, el material de todas las prácticas en la cuenta de usuario.

Fontes de información	
Bibliografía básica	- M. A. Weiss (1995). Estructuras de Datos y Algoritmos. Addison Wesley - G. Brassard y P. Bratley (1997). Fundamentos de Algoritmia. Prentice Hall - U. Manber (1989). Introduction to Algorithms - A Creative Approach. Addison Wesley

Bibliografía complementaria	- R. Sedgewick (1988). Algorithms. Addison Wesley - R. Peña Marí (2005). Diseño de Programas. Formalismo y Abstracción. Tercera edición.. Pearson Prentice Hall - 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

Recomendacións
Materias que se recomenda ter cursado previamente
Estrutura de Datos e da Información/614111102 Matemática Discreta I/614111107 Programación/614111109
Materias que se recomenda cursar simultaneamente
Materias que continúan o temario
Observacións

(*)A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías