

270012 - EDA - Estructuras de Datos y Algoritmos

Unidad responsable: 270 - FIB - Facultad de Informática de Barcelona
Unidad que imparte: 723 - CS - Departamento de Ciencias de la Computación
Curso: 2016
Titulación: GRADO EN INGENIERÍA INFORMÁTICA (Plan 2010). (Unidad docente Obligatoria)
GRADO EN INGENIERÍA FÍSICA (Plan 2011). (Unidad docente Optativa)
Créditos ECTS: 6 Idiomas docencia: Catalán, Castellano, Inglés

Profesorado

Responsable: - Enric Rodríguez Carbonell (erodri@cs.upc.edu)
Otros: - Albert Atserias Peri (atserias@cs.upc.edu)
- Albert Oliveras Lluell (oliveras@cs.upc.edu)
- Alberto Moreno Vega (albertomv@cs.upc.edu)
- Alexandre Vidal Obiols (alexandre.vidal.obiols@est.fib.upc.edu)
- Antoni Lozano Bojados (antoni@cs.upc.edu)
- Conrado Martínez Parra (conrado@cs.upc.edu)
- Jordi Petit Silvestre (jpetit@cs.upc.edu)
- Salvador Roura Ferret (roura@cs.upc.edu)

Capacidades previas

Se espera que los estudiantes estén familiarizados con las técnicas de programación imperativa basada en objetos:

- paso de parámetros,
- clases,
- objetos,
- métodos,
- punteros,
- memoria dinámica,
- genericidad,
- recursividad,
- uso de clases estándar,
- iteradores.

También se espera que conozcan bien al menos un lenguaje imperativo orientado a objetos, preferentemente C++.

Son también requisitos la capacidad crítica y la madurez matemática.

Requisitos

- Pre-requisito PRO2
- Pre-requisito PRO1

Competencias de la titulación a las cuales contribuye la asignatura

Específicas:

- CT2.3. Diseñar, desarrollar, seleccionar y evaluar aplicaciones, sistemas y servicios informáticos, y al mismo tiempo asegurar su fiabilidad, su seguridad y su calidad, conforme a principios éticos y a la legislación y la normativa vigente.
- CT2.4. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los Sistemas de información, incluidos los basados en web.

270012 - EDA - Estructuras de Datos y Algoritmos

CT4.1. Identificar las soluciones algorítmicas más adecuadas para resolver problemas de dificultad mediana.

CT4.2. Razonar sobre la corrección y la eficiencia de una solución algorítmica.

CT4.3. Demostrar conocimiento y capacidad de aplicación de los principios fundamentales y las técnicas básicas de los sistemas inteligentes y su aplicación práctica.

CT5.1. Escoger, combinar y explotar diferentes paradigmas de programación, en el momento de construir software, atendiendo a criterios como la facilidad de desarrollo, la eficiencia, la portabilidad y la mantenibilidad.

CT5.2. Conocer, diseñar y utilizar de forma eficiente los tipos y las estructuras de datos más adecuados para la resolución de un problema.

CT5.3. Diseñar, escribir, probar, depurar, documentar y mantener código en un lenguaje de alto nivel para resolver problemas de programación aplicando esquemas algorítmicos y usando estructuras de datos.

CT5.4. Diseñar la arquitectura de los programas utilizando técnicas de orientación a objetos, de modularización y de especificación e implementación de tipos abstractos de datos.

CT5.5. Usar las herramientas de un entorno de desarrollo de software para crear y desarrollar aplicaciones.

CT8.6. Demostrar comprensión de la importancia de la negociación, de los hábitos de trabajo efectivos, del liderazgo y de las habilidades de comunicación en todos los entornos de desarrollo de software.

CT8.7. Controlar versiones y configuraciones del proyecto.

Genéricas:

G6. USO SOLVENTE DE LOS RECURSOS DE INFORMACIÓN: Gestionar la adquisición, la estructuración, el análisis y la visualización de datos e información del ámbito de la ingeniería informática y valorar de forma crítica los resultados de esta gestión.

Metodologías docentes

El temario se expone de forma muy práctica, a través de la presentación de muchos ejemplos.

Las clases de teoría introducen todos los conceptos y técnicas necesarios, los cuales se ponen en práctica en las clases de problemas y de laboratorio mediante una colección de problemas y ejercicios en un juez automático.

Las dos horas de clases de teoría se hacen semanalmente. Las dos horas de clases de laboratorio se hacen quincenalmente. Las dos horas de clases de problemas se hacen quincenalmente.

La programación del juego integra los conocimientos y las competencias de todo el curso.

El curso utiliza el lenguaje de programación C++.

Objetivos de aprendizaje de la asignatura

1. Comprender las definiciones de las notaciones asintóticas O grande, Omega y Theta y su uso para caracterizar la eficiencia en tiempo y espacio de los algoritmos.

2. Calcular la eficiencia de algoritmos iterativos aplicando las reglas de cálculo adecuadas.

3. Describir la eficiencia de los algoritmos recursivos utilizando recurrencias y comprender y aplicar los teoremas maestros para resolverlas.

4. Diseñar algoritmos para resolver problemas diversos de dificultad media con restricciones tanto de tiempo como de espacio.

5. Comparar la eficiencia de distintos algoritmos para resolver un mismo problema y seleccionar el más adecuado.

6. Conocer, explicar, diseñar, analizar, comparar e implementar algoritmos (como pueden ser mergesort, quicksort, Karatsuba y Strassen, entre otros) utilizando la técnica de dividir y vencer.

270012 - EDA - Estructuras de Datos y Algoritmos

7. Conocer, explicar, diseñar, analizar, comparar e implementar las principales estructuras de datos que se pueden utilizar para implementar diccionarios (tablas, tablas ordenadas, listas, listas ordenadas, tablas de dispersión, árboles binarios de búsqueda, árboles AVL).
8. Conocer, explicar, diseñar, analizar, comparar e implementar las principales estructuras de datos que se pueden utilizar para implementar colas de prioridad (árboles, heaps).
9. Conocer, explicar, diseñar, analizar, comparar e implementar algoritmos que resuelvan problemas clásicos en grafos como recorridos, ordenación topológica y caminos mínimos entre otros.
10. Conocer, explicar, diseñar, analizar, comparar e implementar algoritmos de búsqueda exhaustiva utilizando la técnica de backtracking.
11. Tomar conciencia de los límites de la computación: comprender las implicaciones de la pregunta " $P = NP?$ ", entender el enunciado del Teorema de Cook-Levin, reconocer e identificar varios problemas NP-completos clásicos.
12. Completar y modificar implementaciones, en lenguaje de programación C++, de varios algoritmos para resolver problemas de dificultad media.
13. Identificar y proponer soluciones para problemas de eficiencia de algoritmos y de programas escritos en lenguaje de programación C++.
14. Analizar un juego estratégico para diseñar y programar un jugador eficaz, eficiente, colaborativo y competitivo que maximice las posibilidades de ganar el juego y que sea capaz de establecer alianzas y de coordinarse con otros jugadores.
15. Ejecutar estrategias de búsqueda de información (referencias bibliográficas, artículos científicos, patentes, recursos web de calidad ...) para elaborar un documento que describa, para un problema dado, un algoritmo conocido que lo resuelva, estructurándolo correctamente, citando adecuadamente las fuentes utilizadas y haciendo un uso ético de la información recopilada.
16. Calcular el coste de un algoritmo en los casos peor, mejor y medio.

Horas totales de dedicación del estudiantado

Dedicación total: 150h	Grupo grande/Teoría:	30h	20.00%
	Grupo mediano/Prácticas:	15h	10.00%
	Grupo pequeño/Laboratorio:	15h	10.00%
	Actividades dirigidas:	6h	4.00%
	Aprendizaje autónomo:	84h	56.00%

270012 - EDA - Estructuras de Datos y Algoritmos

Contenidos

Análisis de Algoritmos

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Coste en tiempo y espacio. Caso peor, mejor y medio. Notación asintótica. Análisis del coste de algoritmos iterativos y recursivos.

Divide y Vencerás

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Principios: partición en subproblemas, recombinación de soluciones. Ejemplos: mergesort, quicksort, algoritmo de Karatsuba para multiplicar números grandes, algoritmo de Strassen para multiplicar matrices.

Diccionarios

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Operaciones de diccionarios y diccionarios ordenados. Implementaciones básicas: tablas y listas. Implementaciones avanzadas: tablas de dispersión, árboles binarios de búsqueda, árboles AVL.

Colas con Prioridades

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Operaciones de colas con prioridades. Implementaciones con heaps. Heapsort.

Grafos

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Representaciones: matrices de adyacencia, listas de adyacencia e implícita. Búsqueda en profundidad (DFS). Búsqueda en anchura (BFS). Ordenación topológica. Algoritmo de Dijkstra para caminos mínimos. Algoritmo de Prim para árboles de expansión mínimos.

Generación y Búsqueda Exhaustiva

Competencias de la titulación a las que contribuye el contenido:

270012 - EDA - Estructuras de Datos y Algoritmos

Descripción:

Principios: espacio de soluciones, soluciones parciales, poda. Generación de subconjuntos y permutaciones.

Ejemplos: mochila, viajante de comercio.

Nociones de Intractabilidad

Competencias de la titulación a las que contribuye el contenido:

Descripción:

Introducción básica a las clases P y NP, al Teorema de Cook-Levin, a las reducciones y a la NP-completitud.

270012 - EDA - Estructuras de Datos y Algoritmos

Planificación de actividades

<p>Análisis de Algoritmos</p>	<p>Dedicación: 16h Grupo grande/Teoría: 4h Grupo mediano/Prácticas: 2h Grupo pequeño/Laboratorio: 2h Actividades dirigidas: 0h Aprendizaje autónomo: 8h</p>
<p>Descripción: Desarrollo del tema 1 de la asignatura.</p> <p>Objetivos específicos: 1, 2, 3, 4, 5, 16</p>	
<p>Dividir y Vencer</p>	<p>Dedicación: 16h Grupo grande/Teoría: 4h Grupo mediano/Prácticas: 2h Grupo pequeño/Laboratorio: 2h Actividades dirigidas: 0h Aprendizaje autónomo: 8h</p>
<p>Descripción: Desarrollo del tema 2 de la asignatura.</p> <p>Objetivos específicos: 3, 4, 5, 6, 12, 13</p>	
<p>Diccionarios</p>	<p>Dedicación: 16h Grupo grande/Teoría: 4h Grupo mediano/Prácticas: 2h Grupo pequeño/Laboratorio: 2h Actividades dirigidas: 0h Aprendizaje autónomo: 8h</p>
<p>Descripción: Desarrollo del tema 3 de la asignatura.</p> <p>Objetivos específicos: 4, 5, 7, 12, 13</p>	
<p>Examen parcial de papel</p>	<p>Dedicación: 6h Actividades dirigidas: 2h Aprendizaje autónomo: 4h</p>
<p>Descripción: Se evalúan los objetivos correspondientes a los contenidos 1 y 2.</p> <p>Objetivos específicos: 1, 2, 3, 4, 5, 6, 12</p>	

270012 - EDA - Estructuras de Datos y Algoritmos

Colas de Prioridades	Dedicación: 8h Grupo grande/Teoría: 2h Grupo mediano/Prácticas: 1h Grupo pequeño/Laboratorio: 1h Actividades dirigidas: 0h Aprendizaje autónomo: 4h
Descripción: Desarrollo del tema 4 de la asignatura. Objetivos específicos: 4, 5, 8, 12, 13	
Grafos	Dedicación: 16h Grupo grande/Teoría: 4h Grupo mediano/Prácticas: 2h Grupo pequeño/Laboratorio: 2h Actividades dirigidas: 0h Aprendizaje autónomo: 8h
Descripción: Desarrollo del tema 5 de la asignatura. Objetivos específicos: 4, 5, 9, 12, 13	
Juego	Dedicación: 8h Actividades dirigidas: 0h Aprendizaje autónomo: 8h

270012 - EDA - Estructuras de Datos y Algoritmos

Descripción:

Se evalúan los objetivos correspondientes al objetivo específico 15.

Se publicará un enunciado que consistirá en la descripción de un juego de estrategia. Los estudiantes deberán programar un jugador para este juego (es decir, implementar una estrategia para intentar ganar el juego).

Se llevará a cabo una competición en la que los estudiantes competirán los unos contra los otros, de la que se obtendrá una clasificación. Para participar en esta competición, los jugadores de los estudiantes deberán superar un test de cualificación.

La nota correspondiente a esta parte se calculará a partir de la posición en la clasificación de forma proporcional, garantizando que el campeón tenga un 10 y que todos los participantes que tengan un jugador cualificado tengan una nota mínima de 5. Los estudiantes que no hayan conseguido ningún jugador cualificado tendrán una nota de 0.

Si se detecta alguna acción fraudulenta, la nota correspondiente a esta parte será restada de (en lugar de sumada a) la nota del curso.

Objetivos específicos:

14

Generación y Búsqueda Exhaustiva

Dedicación: 18h

Grupo grande/Teoría: 4h

Grupo mediano/Prácticas: 2h

Grupo pequeño/Laboratorio: 4h

Actividades dirigidas: 0h

Aprendizaje autónomo: 8h

Descripción:

Desarrollo del tema 6 de la asignatura.

Objetivos específicos:

4, 5, 10, 12, 13

Nociones de Intratabilidad y indecidibilidad

Dedicación: 10h

Grupo grande/Teoría: 4h

Grupo mediano/Prácticas: 2h

Grupo pequeño/Laboratorio: 0h

Actividades dirigidas: 0h

Aprendizaje autónomo: 4h

Descripción:

Desarrollo del tema 7 de la asignatura.

Objetivos específicos:

11

270012 - EDA - Estructuras de Datos y Algoritmos

Examen parcial de ordenador	Dedicación: 6h 30m Actividades dirigidas: 2h 30m Aprendizaje autónomo: 4h
<p>Descripción: Se evalúa la vertiente de laboratorio, es decir, de implementación, de los contenidos tratados hasta la fecha del examen.</p> <p>Frente al ordenador, los estudiantes reciben dos o tres problemas. Un problema se define con un enunciado, uno o más juegos de pruebas públicos y, quizá, cierto código ya implementado. Cuando un estudiante quiera entregar su solución para alguno de los problemas, la enviará a un juez automático que, en pocos segundos, le devolverá el veredicto sobre el comportamiento de su programa. El estudiante puede reenviar hasta 10 soluciones para el mismo problema. Los profesores corregirán el último envío realizado para cada problema.</p> <p>Objetivos específicos: 7, 8, 9, 10</p>	
Consolidación.	Dedicación: 12h 30m Grupo grande/Teoría: 4h Grupo mediano/Prácticas: 2h Grupo pequeño/Laboratorio: 2h Actividades dirigidas: 0h Aprendizaje autónomo: 4h 30m
<p>Descripción: Repaso.</p> <p>Objetivos específicos: 1, 2, 3, 4, 5, 12, 13</p>	
Tutoriales Bibliotècnica-BRGF.	Dedicación: 2h 30m Grupo grande/Teoría: 0h Grupo mediano/Prácticas: 0h Grupo pequeño/Laboratorio: 0h Actividades dirigidas: 0h Aprendizaje autónomo: 2h 30m
<p>Descripción: Autoaprendizaje mediante tutoriales de la Bibliotècnica de la BRGF sobre: &#8203;&#8203;la propiedad intelectual, el uso ético de la información y el uso de software de gestión de referencias bibliográficas.</p> <p>Objetivos específicos: 15</p>	

270012 - EDA - Estructuras de Datos y Algoritmos

Práctica de uso solvente de recursos de información.	Dedicación: 3h Actividades dirigidas: 0h Aprendizaje autónomo: 3h
<p>Descripción:</p> <p>Se evalúan los objetivos correspondientes al objetivo específico 16.</p> <p>Se publicará un enunciado que consistirá en la descripción de un problema computacional y el nombre de un algoritmo que lo resuelve.</p> <p>Los estudiantes tendrán que buscar (en la biblioteca, en la web, ...) información sobre el problema y el algoritmo y elaborar un documento breve y bien estructurado que incluya correctamente las fuentes consultadas.</p> <p>El documento deberá entregarse el día del examen final.</p> <p>La competencia transversal se evaluará en base a este documento.</p> <p>Objetivos específicos:</p> <p>15</p>	
Examen final	Dedicación: 13h Actividades dirigidas: 3h Aprendizaje autónomo: 10h
<p>Descripción:</p> <p>Se evalúa los objetivos correspondientes a los contenidos 1 a 7.</p> <p>Objetivos específicos:</p> <p>1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13</p>	

Sistema de calificación

NPP = nota del examen parcial de papel (entre 0 y 10)

NPO = nota del examen parcial de ordenador (entre 0 y 10)

NF = nota del examen final (entre 0 y 10)

NJ = nota del juego (entre 0 y 10)

NOTA = $\min(10, \max(30\%NPP + 30\%NPO + 30\%NF + 20\%NJ, 30\%NPO + 60\%NF + 20\%NJ))$

270012 - EDA - Estructuras de Datos y Algoritmos

Bibliografía

Básica:

Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C. Introduction to algorithms. 3rd ed. MIT Press, 2009. ISBN 9780262033848.

Brassard, G.; Bratley, P. Fundamentos de algoritmia. Prentice Hall, 1997. ISBN 848966000X.

Weiss, M.A. Data structures and algorithm analysis in C++. 4th ed int. Pearson, 2014. ISBN 0273769383.

Kreher, D. L.; Stinson, D. R. Combinatorial algorithms : generation, enumeration and search. CRC Press, 1999. ISBN 084933988X.

Garey, M. R; Johnson, D. S. Computers and intractability : a guide to the theory of NP-Completeness. W.H. Freeman, 1979. ISBN 0716710447.

Stroustrup, B. The C++ programming language. 4th ed. Addison-Wesley, 2013. ISBN 9780321563842.

Neapolitan, Richard E. Foundations of algorithms. 5th ed. Jones and Bartlett Learning, cop. 2015. ISBN 9781284049190.

Complementaria:

Manber, U. Introduction to algorithms: a creative approach. Repr. with corr. Addison-Wesley, 1989. ISBN 0201120372.

Harel, D.; Feldman, Y. Algorithmics: the spirit of computing. 3rd ed. Addison-Wesley, 2004. ISBN 0 321 11784 0.

Dasgupta, S.; Papadimitriou, Ch.; Vazirani, U. Algorithms. Mc Graw Hill Higher Education, 2008. ISBN 9780073523408.

Sedgewick, R. Algorithms in C++. 3rd ed. Addison-Wesley, 1998-2002. ISBN 9780201350883.

Sedgewick, Robert; Wyk, Christopher J. Van. Algorithms in C++ (Vol. 2). 3rd ed. Addison-Wesley, 1998-2002. ISBN 9780201361186.

Otros recursos:

Enlace web

<http://ocw.mit.edu/courses/#electrical-engineering-and-computer-science>

<http://www.cs.pitt.edu/~kirk/algorithmcourses/index.html>

<https://www.judge.org/>

<http://www.cs.sunysb.edu/~algorithm/>

<http://uva.onlinejudge.org/>

<http://www.topcoder.com>