

Guía docente

270213 - AP3 - Algoritmia y Programación 3

Última modificación: 10/07/2025

Unidad responsable: Facultad de Informática de Barcelona
Unidad que imparte: 723 - CS - Departamento de Ciencias de la Computación.
Titulación: GRADO EN CIENCIA E INGENIERÍA DE DATOS (Plan 2017). (Asignatura obligatoria).
Curso: 2025 **Créditos ECTS:** 6.0 **Idiomas:** Catalán

PROFESORADO

Profesorado responsable: ENRIC RODRIGUEZ CARBONELL
Otros: Primer quadrimestre:
ALBERT OLIVERAS LLUNELL - 11, 12, 13
ENRIC RODRIGUEZ CARBONELL - 11, 12, 13

CAPACIDADES PREVIAS

- Familiaridad con las técnicas básicas de programación: iteraciones, alternativas, funciones recursivas, paso de parámetros, clases, objetos, métodos, ...
- Conocimiento de conceptos algorítmicos básicos: eficiencia de algoritmos, notación asintótica, grafos, recorrido en grafos, estructuras de datos (listas, árboles de búsqueda, hash, heaps, ...)
- Conocimientos básicos de matemática discreta, álgebra lineal y cálculo
- Conocimientos básicos de teoría de la probabilidad y estadística

COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

Específicas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.
CE7. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los datos.

Genéricas:

CG1. Concebir sistemas computacionales que integren datos de procedencias y formas muy diversas, creen con ellos modelos matemáticos, razonen sobre dichos modelos y actúen en consecuencia, aprendiendo de la experiencia.
CG2. Elegir y aplicar los métodos y técnicas más adecuados a un problema definido por datos que representen un reto por su volumen, velocidad, variedad o heterogeneidad, incluidos métodos informáticos, matemáticos, estadísticos y de procesamiento de la señal.
CG5. Poder recurrir a conocimientos fundamentales y metodologías de trabajo sólidas adquiridos durante los estudios para adaptarse a los nuevos escenarios tecnológicos del futuro.

Transversales:

CT4. Trabajo en equipo. Ser capaz de trabajar como miembro de un equipo interdisciplinar, ya sea como un miembro más o realizando tareas de dirección, con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos teniendo en cuenta los recursos disponibles.

CT5. Uso solvente de los recursos de información. Gestionar la adquisición, la estructuración, el análisis y la visualización de datos e información en el ámbito de especialidad y valorar de forma crítica los resultados de dicha gestión.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

CT7. Tercera lengua. Conocer una tercera lengua, preferentemente el inglés, con un nivel adecuado oral y escrito y en consonancia con las necesidades que tendrán los titulados y tituladas.

Básicas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

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 posen 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.

El proyecto integra los conocimientos y las competencias de todo el curso.

OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

1. 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 y identificar varios problemas NP-completos clásicos.

2. Conocer, explicar, diseñar, analizar, comparar e implementar algoritmos de búsqueda exhaustiva usando la técnica de backtracking.

3. Conocer el esquema de programación dinámica, identificar cuándo se puede aplicar y cómo, y familiarizarse con algunos algoritmos de programación dinámica fundamentales.

4. Conocer el esquema de los algoritmos voraces, identificar cuándo se pueden aplicar y cómo, conocer las técnicas más habituales para demostrar su corrección y familiarizarse con algunos algoritmos voraces fundamentales.

5. Completar y modificar implementaciones, en el lenguaje de programación Python, de varios algoritmos para resolver problemas de dificultad media.

6. Identificar y proponer soluciones a posibles problemas de eficiencia en programas escritos en el lenguaje de programación Python.

7. Desarrollar proyectos de tamaño medio como miembro de un equipo, aprendiendo así a dividir un proyecto en partes pequeñas, distribuir las entre sus miembros y actuar con responsabilidad de forma coordinada para la correcta finalización de las tareas asignadas.

9. Conocer algoritmos basados en búsqueda local para resolver eficientemente problemas intratables. Conocer una variedad de metaheurísticas de diferente naturaleza y ser capaz de identificar cuándo y cómo se pueden aplicar en problemas concretos computacionalmente duros.

10. Conocer los fundamentos de autómatas finitos y expresiones regulares para poderlos usar en la práctica (búsqueda de patrones en textos, etc.).

HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTE

Tipo	Horas	Porcentaje
Horas grupo pequeño	30,0	20.00
Horas aprendizaje autónomo	90,0	60.00
Horas grupo grande	30,0	20.00

Dedicación total: 150 h

CONTENIDOS

Tractabilidad: clases de problemas P y NP

Descripción:

Clases P y NP, Teorema de Cook-Levin, reducciones, NP-completitud.

Búsqueda exhaustiva

Descripción:

Principios teóricos: espacio de soluciones, soluciones parciales, poda. Ejemplos: subconjuntos, permutaciones, viajante de comercio, suma de subconjuntos.

Programación dinámica

Descripción:

Esquema top-down (memoización). Esquema bottom-up (tabulación). Ejemplos: Fibonacci, números binomiales, mochila, multiplicación de secuencias matrices.

Algoritmos voraces

Descripción:

Principios teóricos: esquema general de los algoritmos voraces. Ejemplos: planificación de tareas, etc. mínimos.

Metaheurísticas

Descripción:

Búsqueda local. Simulated Annealing, Tabu Search, GRASP, algoritmos genéticos.

Autómatas finitos y expresiones regulares

Descripción:

Alfabetos, palabras, lenguajes. Autómatas finitos deterministas, autómatas finitos indeterministas, autómatas finitos con lambda-transiciones, equivalencia entre modelos de autómatas, minimización de autómatas. Expresiones regulares, equivalencia con autómatas. Operaciones.

ACTIVIDADES

Tractabilidad

Objetivos específicos:

1

Competencias relacionadas:

CG5. Poder recurrir a conocimientos fundamentales y metodologías de trabajo sólidas adquiridos durante los estudios para adaptarse a los nuevos escenarios tecnológicos del futuro.

Dedicación: 18h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 6h

Grupo mediano/Prácticas: 4h

Búsqueda Exhaustiva

Objetivos específicos:

2, 5, 6

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 16h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Grupo pequeño/Laboratorio: 4h

Programación Dinámica

Objetivos específicos:

3, 5, 6

Competencias relacionadas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 16h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Grupo pequeño/Laboratorio: 4h

Algoritmos Voraces

Objetivos específicos:

4, 5, 6

Competencias relacionadas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 16h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Grupo mediano/Prácticas: 2h

Grupo pequeño/Laboratorio: 2h

Metaheurísticas

Objetivos específicos:

5, 9

Competencias relacionadas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 18h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Grupo pequeño/Laboratorio: 6h

Autómatas finitos y expresiones regulares

Objetivos específicos:

10

Competencias relacionadas:

CE7. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los datos.

Dedicación: 16h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Grupo mediano/Prácticas: 4h

Consolidación

Objetivos específicos:

1, 2, 3, 4, 9, 10

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CE7. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los datos.

CG5. Poder recurrir a conocimientos fundamentales y metodologías de trabajo sólidas adquiridos durante los estudios para adaptarse a los nuevos escenarios tecnológicos del futuro.

Dedicación: 12h

Aprendizaje autónomo: 8h

Grupo grande/Teoría: 4h

Examen parcial

Objetivos específicos:

1, 2, 3

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CG5. Poder recurrir a conocimientos fundamentales y metodologías de trabajo sólidas adquiridos durante los estudios para adaptarse a los nuevos escenarios tecnológicos del futuro.

Dedicación: 3h

Actividades dirigidas: 3h

Proyecto - Búsqueda Exhaustiva

Objetivos específicos:

2, 5, 6, 7

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CG2. Elegir y aplicar los métodos y técnicas más adecuados a un problema definido por datos que representen un reto por su volumen, velocidad, variedad o heterogeneidad, incluidos métodos informáticos, matemáticos, estadísticos y de procesamiento de la señal.

CG1. Concebir sistemas computacionales que integren datos de procedencias y formas muy diversas, creen con ellos modelos matemáticos, razonen sobre dichos modelos y actúen en consecuencia, aprendiendo de la experiencia.

CT4. Trabajo en equipo. Ser capaz de trabajar como miembro de un equipo interdisciplinar, ya sea como un miembro más o realizando tareas de dirección, con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos teniendo en cuenta los recursos disponibles.

CT5. Uso solvente de los recursos de información. Gestionar la adquisición, la estructuración, el análisis y la visualización de datos e información en el ámbito de especialidad y valorar de forma crítica los resultados de dicha gestión.

CT7. Tercera lengua. Conocer una tercera lengua, preferentemente el inglés, con un nivel adecuado oral y escrito y en consonancia con las necesidades que tendrán los titulados y tituladas.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 9h

Aprendizaje autónomo: 9h

Proyecto - Algoritmos Voraces

Objetivos específicos:

4, 5, 6, 7

Competencias relacionadas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CG2. Elegir y aplicar los métodos y técnicas más adecuados a un problema definido por datos que representen un reto por su volumen, velocidad, variedad o heterogeneidad, incluidos métodos informáticos, matemáticos, estadísticos y de procesamiento de la señal.

CG1. Concebir sistemas computacionales que integren datos de procedencias y formas muy diversas, creen con ellos modelos matemáticos, razonen sobre dichos modelos y actúen en consecuencia, aprendiendo de la experiencia.

CT4. Trabajo en equipo. Ser capaz de trabajar como miembro de un equipo interdisciplinar, ya sea como un miembro más o realizando tareas de dirección, con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos teniendo en cuenta los recursos disponibles.

CT5. Uso solvente de los recursos de información. Gestionar la adquisición, la estructuración, el análisis y la visualización de datos e información en el ámbito de especialidad y valorar de forma crítica los resultados de dicha gestión.

CT7. Tercera lengua. Conocer una tercera lengua, preferentemente el inglés, con un nivel adecuado oral y escrito y en consonancia con las necesidades que tendrán los titulados y tituladas.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 9h

Aprendizaje autónomo: 9h

Examen de laboratorio

Objetivos específicos:

2, 3, 4, 5, 6, 9, 10

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CE7. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los datos.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 3h

Actividades dirigidas: 3h

Proyecto - Metaheurísticas

Objetivos específicos:

5, 6, 7, 9

Competencias relacionadas:

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CG2. Elegir y aplicar los métodos y técnicas más adecuados a un problema definido por datos que representen un reto por su volumen, velocidad, variedad o heterogeneidad, incluidos métodos informáticos, matemáticos, estadísticos y de procesamiento de la señal.

CG1. Concebir sistemas computacionales que integren datos de procedencias y formas muy diversas, creen con ellos modelos matemáticos, razonen sobre dichos modelos y actúen en consecuencia, aprendiendo de la experiencia.

CT4. Trabajo en equipo. Ser capaz de trabajar como miembro de un equipo interdisciplinar, ya sea como un miembro más o realizando tareas de dirección, con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos teniendo en cuenta los recursos disponibles.

CT5. Uso solvente de los recursos de información. Gestionar la adquisición, la estructuración, el análisis y la visualización de datos e información en el ámbito de especialidad y valorar de forma crítica los resultados de dicha gestión.

CT7. Tercera lengua. Conocer una tercera lengua, preferentemente el inglés, con un nivel adecuado oral y escrito y en consonancia con las necesidades que tendrán los titulados y tituladas.

CT6. Aprendizaje autónomo. Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar dicho conocimiento.

Dedicación: 10h

Aprendizaje autónomo: 10h

Examen final

Objetivos específicos:

1, 2, 3, 4, 9, 10

Competencias relacionadas:

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CE2. Ser capaz de programar soluciones a problemas de ingeniería: Diseñar soluciones algorítmicas eficientes a un problema computacional dado, implementarlas en forma de Programa robusto, estructurado y mantenible, y comprobar la validez de la solución.

CE7. Demostrar conocimiento y capacidad de aplicación de las herramientas necesarias para el almacenaje, el procesamiento y el acceso a los datos.

CG5. Poder recurrir a conocimientos fundamentales y metodologías de trabajo sólidas adquiridos durante los estudios para adaptarse a los nuevos escenarios tecnológicos del futuro.

Dedicación: 3h

Actividades dirigidas: 3h

SISTEMA DE CALIFICACIÓN

NPar = nota examen parcial

NFT = nota examen final de teoría

NFL = nota examen final de laboratorio

NPro = nota proyecto

NOTA FINAL = $\max(30\%NPar + 30\%NFT + 20\%NFL + 20\%NPro, 60\%NFT + 20\%NFL + 20\%NPro)$

La nota del examen de reevaluación, si hay y es más alta, sustituye la nota del examen final de teoría (NFT). Las notas de parcial, proyecto y laboratorio (NPar, NPro, NFL) se conservan.

BIBLIOGRAFÍA

Básica:

- Cormen, T.H. [et al.]. Introduction to algorithms [en línea]. Fourth edition. Cambridge, Massachusetts: MIT Press, [2022] [Consulta: 16/07/2025]. Disponible a: <https://research-ebsco-com.recursos.biblioteca.upc.edu/c/ik5pvi/search/details/lq2sgumirf?db=nlebk>. ISBN 9780262046305.
- Garey, M.R.; Johnson, D.S. Computers and intractability: a guide to the theory of NP-Completeness. W.H. Freeman, 1979. ISBN 0716710447.
- Neapolitan, R.E. Foundations of algorithms. 5th ed. Jones and Bartlett Learning, 2015. ISBN 9781284049190.
- Siarry, P. (ed.). Metaheuristics. 5th ed. Springer, 2017. ISBN 9783319832845.
- Cases, R.; Màrquez, L. Llenguatges, gramàtiques i autòmats: curs bàsic. 2a ed. Barcelona: Edicions UPC, 2003. ISBN 8483017288.

Complementaria:

- Gendreau, M.; Potvin, J.-Y. Handbook of metaheuristics. 3rd ed. New York: Springer, 2018. ISBN 9783319910857.
- Dasgupta, S.; Papadimitriou, C.; Vazirani, U.. Algorithms. Boston: Mc Graw Hill Higher Education, 2008. ISBN 9780073523408.
- Sedgewick, R.; Wayne, K. Algorithms. 4th ed. Upper Saddle River, NJ: Addison-Wesley, 2011. ISBN 9780321573513.
- Kleinberg, J.; Tardos, É. Algorithm design. New int. ed. Pearson, 2014. ISBN 9781292023946.
- Hopcroft, J.E.; Motwani, R.; Ullman, J.D. Introduction to automata theory, languages, and computation. 3rd ed. Boston: Pearson/Addison Wesley, 2007. ISBN 0321462254.

RECURSOS

Enlace web:

- <https://jutge.org/>