

**40962 - ALGORITMOS Y
PROGRAMACIÓN**

CENTRO: 180 - *Escuela de Ingeniería Informática*

TITULACIÓN: 4008 - *Grado en Ingeniería Informática*

ASIGNATURA: 40962 - *ALGORITMOS Y PROGRAMACIÓN*

CÓDIGO UNESCO: **TIPO:** *Obligatoria* **CURSO:** 2 **SEMESTRE:** 1º semestre

CRÉDITOS ECTS: 6 **Especificar créditos de cada lengua:** **ESPAÑOL:** 6 **INGLÉS:**

SUMMARY

Algorithms and Programming is a subject common to the branch of Computer Engineering which belongs to the Software Development Engineering module. The contents of this course complement and extend the subjects of Programming Foundations I and II that are taught respectively in the first and second semesters of the first year; and together with the second-year subjects "Programming Technologies" and "Data Structures and Programming" completes the basic knowledge on Algorithms and Programming.

REQUISITOS PREVIOS

Se recomienda haber cursado:

- Fundamentos de Programación I
- Fundamentos de Programación II
- Matemática Discreta

Plan de Enseñanza (Plan de trabajo del profesorado)

Contribución de la asignatura al perfil profesional:

La asignatura de Algoritmos y Programación adscrita al módulo de Ingeniería y Desarrollo de Software es una asignatura común a la rama de la Ingeniería Informática que el estudiante debe cursar obligatoriamente para cubrir el perfil profesional de la titulación. Amplía las asignaturas de Fundamentos de Programación I y II que se imparten respectivamente en el primer y segundo semestre de primer curso y, junto con las asignaturas de segundo curso "Tecnologías de Programación" y "Estructuras de Datos y Programación", completa la materia de Programación. Complementa asimismo la asignatura básica de rama de primer curso "Matemática Discreta".

Algoritmos y Programación, por un lado, agrega técnicas de diseño algorítmico, contribuyendo de esta forma en la abundancia de la capacidad de seleccionar la más adecuada para resolver un problema concreto, y por otro lado proporciona el tránsito desde los algoritmos que resuelven con creciente eficacia un determinado problema, hasta considerar algoritmos alternativos que lo resuelven más eficazmente, introduciendo de esta manera el análisis de la eficiencia de los algoritmos y la complejidad computacional.

La asignatura ofrece al estudiante un amplio abanico de algoritmos y técnicas de diseño, teniendo la eficiencia como factor fundamental de calidad y criterio de selección imprescindible en la

ingeniería de desarrollo de software. Como alternativa para la resolución de este tipo de problemas se plantea la programación declarativa y dentro del paraguas que este paradigma de programación permite se refuerza el planteamiento basado en la programación con restricciones, y la programación entera mixta. De este modo, se intensifica la capacidad del estudiante para adaptarse a la resolución de nuevos problemas de programación con creatividad, versatilidad, iniciativa y autonomía.

Competencias que tiene asignadas:

CB1, CB5, G3, G4, G8, G9, N2, N3, CI6, CI8

Objetivos:

OB1: Conocer reglas y propiedades que permiten analizar la eficiencia de algoritmos.

OB2: Conocer técnicas de diseño de algoritmos.

OB3: Familiarizar al estudiante con algoritmos fundamentales y la complejidad computacional de un problema.

OB4: Familiarizar al estudiante con la programación declarativa.

Contenidos:

Tema 1. Análisis de la eficiencia algorítmica [4 horas]

- 1.1 Notaciones asintóticas
- 1.2 Análisis de eficiencia de algoritmos iterativos
- 1.3 Análisis de eficiencia de algoritmos recursivos

Bibliografía: [1, 2]

Tema 2. Estrategias de diseño algorítmico [16 horas]

- 2.1 Fuerza bruta
- 2.2 Ávida (greedy)
- 2.3 Divide y vencerás
- 2.4 Vuelta atrás (backtracking)
- 2.5 Programación dinámica

Bibliografía: [1, 3, 4, 10]

Tema 3. Complejidad computacional [4 horas]

- 3.1 Complejidad computacional de un problema. Problemas P vs NP
- 3.2 Ramificación y Acotación

Bibliografía: [8]

Tema 4. Programación declarativa [12 horas]

- 4.1 Introducción a la programación declarativa. Programación funcional/lógica
- 4.2 Programación con restricciones

Bibliografía: [9]

Tema 5. Algoritmos aproximados. [4 horas]

- 5.1 Algoritmos de aproximación rho-Aproximados
- 5.2 Esquemas de Aproximación de Tiempo Polinómico (1-e)

Bibliografía: [5]

Tema 6. Algoritmos heurísticos y metaheurísticos [12 horas]

6.1 Heurísticas y búsqueda local

6.2 Metaheurísticas

Bibliografía: [6]

Tema 7. Programación entera mixta [8 horas]

7.1 Fundamentos de la Programación Entera Mixta

7.2 Diferencia de modelado entre programación con restricciones y programación entera mixta.

7.3 Planos de corte

Bibliografía: [7]

CONTENIDOS PRÁCTICOS

1. Problema del Agente Viajero. Se presenta en el Tema 1.

2. Problema de la Mochila. Se presenta en el Tema 2.

3. Problema de Coloreado de Grafo. Se presenta en el Tema 4.

Se plantean 3 problemas NP completo durante el transcurso de la asignatura. El estudiante irá aplicando las distintas técnicas de diseño algorítmico y paradigmas de programación a estos 3 problemas tipo.

Metodología:

En el horario establecido por el centro para las sesiones académicas, se utilizan dos modalidades para los contextos de las actividades que el profesor y los estudiantes desarrollarán en la asignatura: a nivel individual y a nivel de grupo.

DESCRIPCIÓN DE LAS ACTIVIDADES FORMATIVAS PRESENCIALES

AF1: Sesiones académicas teóricas o de fundamentación.

El profesor presenta a todo el grupo el contenido de cada tema; transmite a los estudiantes el conocimiento fundamental de la materia por medio de clases expositivas, explicativas y/o demostrativas; publica el material disponible en la página web de la asignatura; presenta el trabajo de exposición en clase a realizar en grupo y establece un período adecuado para su finalización.

En el caso de que se interrumpa la presencialidad académica. Se grabarán vídeos de cada tema que se pondrán a disposición de los estudiantes a través del campus virtual.

AF2: Sesiones académicas de aplicación práctica.

El trabajo práctico en grupo está compuesto de actividades dirigidas por el profesor, que suponen la realización de programas en lenguaje Python, que les permiten aplicar los conocimientos adquiridos en las AF1. El profesor presenta la actividad en las AF2, publica el enunciado en la página web de la asignatura, indica las tareas a realizar de forma individual o en grupo, establece un plazo de entrega y el porcentaje que aporta a la nota del trabajo práctico. El trabajo colaborativo se desarrolla en las AF2 comprendidas en este período de tiempo, incluyendo la discusión de aportaciones individuales y la resolución de dudas entre iguales.

En el caso de que se interrumpa la presencialidad académica, se podrá continuar con la AF2 de forma telemática.

AF3. Exposiciones de trabajos.

Estas sesiones están orientadas a la participación y la interacción de los estudiantes entre sí y de estos con el profesor.

En el caso de que se interrumpa la actividad académica presencial, el estudiante grabará un vídeo con la presentación del trabajo.

DESCRIPCIÓN DE LAS ACTIVIDADES FORMATIVAS NO PRESENCIALES

AF4: Preparación para los exámenes: estudio y asimilación de los contenidos teóricos.

AF5: Preparación de aportaciones individuales al trabajo en equipo relativas a la resolución de problemas.

AF6: Preparación de las aportaciones individuales al trabajo en equipo relativas al desarrollo de las actividades prácticas.

Evaluación:

Criterios de evaluación

CRITERIOS DE EVALUACIÓN

La evaluación de la asignatura se realizará teniendo en cuenta tres aspectos principales: los contenidos teóricos, la realización de las actividades prácticas y la calidad de un trabajo realizado individualmente o en grupo.

Las fuentes y los criterios de evaluación que se emplearán serán los siguientes:

1) Cuestionario de estudiante

Para garantizar que los estudiantes entienden los conceptos principales de los contenidos teóricos, por cada tema deben realizar un 'cuestionario de estudiantes'.

La escala que se utilizará para valorar esta parte es de 0 a 10 puntos [FE1]. La nota final de esta parte será la media geométrica de los resultados obtenidos en los distintos cuestionarios.

Esta fuente de evaluación está relacionada con las actividades formativas AF1 y AF4.

2) Trabajo práctico

2.1) Trabajo práctico individual. Se plantean tareas individuales en el campus virtual relacionados con el contenido concreto de cada tema.

La escala que se utilizará para valorar el trabajo práctico es de 0 a 10 puntos [FE2].

Para cada trabajo se deberán entregar los ficheros y el código fuente, en su caso.

Esta fuente de evaluación está relacionada con las actividades formativas AF1, AF2 y AF5.

2.2) Trabajo Práctico en Grupo. Se realizarán 3 competiciones durante el curso de problemas NP-Completo donde los estudiantes deben aplicar, de entre las técnicas de diseño algorítmico vistas en el curso, aquellas que minimizan el error cuadrático medio con respecto al óptimo en problemas NP-Completo.

Concretamente los problemas de las competiciones son:

- Problema de la mochila
- Coloreado de grafo
- Problema del Agente Viajero

La nota obtenida dependerá de la posición ocupada en la competición, de la variabilidad de las soluciones propuestas, la originalidad y de la estructuración y diseño de las soluciones. Las entregas se realizarán por grupos. La escala que se utilizará para valorar esta parte es de 0 a 10 puntos [FE3].

Esta fuente de evaluación está relacionada con las actividades formativas AF1, AF2 y AF6.

La nota final de la parte práctica será la media geométrica de los resultados obtenidos en el apartado 2.1 y 2.2

3) Exposición del trabajo realizado individualmente o en grupo.

Exposición oral de cada estudiante. El estudiante grabará un vídeo con la presentación, y presentará un informe en formato pdf. El tiempo estimado por estudiante es de 5 minutos de presentación, con un máximo de 15 minutos por trabajo.

La escala que se utilizará para valorar este trabajo es de 0 a 10 puntos [FE4] y su evaluación se efectuará a partir de la exposición del trabajo. El contenido del trabajo aportará el 60% de la nota y la implementación o ejemplos relacionados con el trabajo, en el caso de que sea posible, aportará el 40% de la nota.

Esta fuente de evaluación está relacionada con la actividad formativa AF3.

Sistemas de evaluación

----- 1) Evaluación continua

* Teórica

La escala que se utilizará para valorar la teoría es de 0 a 10 puntos y su evaluación se efectuará a partir de tres FE1 que constarán de preguntas relativas a los temas tratados. Cada examen aportará el mismo porcentaje a la evaluación teórica y se valorará de 0 a 10 puntos.

* Trabajo práctico

La escala que se utilizará para valorar el trabajo práctico es de 0 a 10 puntos y su evaluación se efectuará a partir de las FE2 que se irán planteando regularmente y de la FE3. El enunciado de cada práctica incluirá las especificaciones, plazos de entrega y porcentaje que aporta a la evaluación del trabajo práctico. Para cada trabajo se deberán entregar los ficheros conteniendo el código fuente, utilizando el medio que se establezca dentro de los plazos establecidos.

* Exposición del trabajo realizado en grupos

La escala que se utilizará para valorar el FE4 es de 0 a 10 puntos y su evaluación se efectuará a partir de la exposición del trabajo. El contenido del trabajo aporta el 60% de la nota y la exposición oral el 40% restante.

2) Sistema de evaluación de las convocatorias extraordinaria y especial

Para la convocatoria extraordinaria y especial se usarán los mismos sistemas de evaluación que para la convocatoria ordinaria.

Criterios de calificación

1) Evaluación continua (Convocatoria ordinaria):

Para superar la asignatura habrá que obtener una nota mínima de 5 puntos en la evaluación de: la teoría, el trabajo práctico en grupos y la exposición del trabajo realizado en grupos. La nota final se calcula con la siguiente fórmula:

$$NF = NT * 0.5 + TPI * 0.15 + TPG * 0.25 + NET * 0.1$$

Donde NF representa la nota final, NT la nota de teoría, TPI la nota del trabajo práctico individual, TPG la nota de trabajo práctico en grupo y NET la nota de la exposición del trabajo.

Para aplicar la fórmula es necesario sacar al menos un 4 en cada una de las partes.

La nota en acta de los estudiantes que no cumplan las condiciones establecidas para superar la asignatura será un suspenso con un valor numérico máximo de 4.

2) Convocatorias extraordinaria y especial:

Para la convocatoria extraordinaria y especial se usarán los mismos criterios de calificación que para la convocatoria ordinaria.

Plan de Aprendizaje (Plan de trabajo de cada estudiante)

Tareas y actividades que realizará según distintos contextos profesionales (científico, profesional, institucional, social)

Las tareas que realizará el estudiante, todas ellas pertenecientes a los contextos científico y profesional, son las siguientes:

Ta1. Búsqueda, consulta y estudio de la documentación específica y técnica de la materia.

Ta2. Preparación de la exposición del trabajo.

Ta3. Trabajo práctico consistente en la realización de las prácticas de laboratorio que requieren solución individual o en grupo.

Ta4. Exámenes escritos.

Temporalización semanal de tareas y actividades (distribución de tiempos en distintas actividades y en presencialidad - no presencialidad)

Los estudiantes dedicarán dos horas semanales presenciales a las AF1 y otras dos a las sesiones académicas de aplicación práctica, que incluirán la realización de ejercicios de aplicación de las técnicas introducidas en la teoría y la solución de problemas de programación (60 horas en 15 semanas). Los exámenes que se realizarán a lo largo del semestre consumen 6 horas presenciales en total. Asimismo, cada grupo consumirá cinco horas (no presenciales) en la elaboración de la exposición del trabajo solicitado.

En promedio, realizarán cada semana 5 horas de trabajo no presencial dedicado a la consulta y asimilación de materiales documentales (2 horas en promedio) y la realización de ejercicios y trabajo práctico (3 horas en promedio), que suponen 75 horas en 15 semanas.

Las 4 horas restantes las distribuirá el estudiante, en función de sus necesidades particulares, entre atención tutorial individual, tiempo adicional de estudio y realización de ejercicios y trabajos.

Recursos que tendrá que utilizar adecuadamente en cada uno de los contextos profesionales.

Re1. Bibliografía, documentación específica sobre el contenido de la asignatura y documentación técnica. Contextos: científico y profesional.

Re2. Navegador web (búsquedas, consultas de documentación electrónica, interacción con compañeros de grupo). Contextos: científico, profesional y social.

Re3. Entorno Integrado de Desarrollo para Python. Contextos: científico y profesional.

Re4. Herramientas proporcionadas por el Campus Virtual. Contextos: científico, profesional y social.

Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas.

RA1: Calcular la eficiencia de los algoritmos en términos de complejidad computacional. Este resultado de aprendizaje se adquiere con las actividades formativas AF1, AF2, AF3, AF4, AF5 y AF6

RA2: Diseñar soluciones a problemas aplicando o adaptando estrategias conocidas. Este resultado de aprendizaje se adquiere con las actividades formativas AF1, AF2, AF4 y AF5.

RA3: Desarrollar programas aplicando modelos de programación declarativa. Este resultado de aprendizaje se adquiere con las actividades formativas AF1, AF2, AF4 y AF5.

Plan Tutorial

Atención presencial individualizada (incluir las acciones dirigidas a estudiantes en 5ª, 6ª y 7ª convocatoria)

Se usará para aclarar dudas, asesorar al estudiante en las tareas individuales y optimizar su rendimiento. El profesor atenderá consultas del alumnado en su horario de tutorías. Las citas se pueden concertar a través del correo electrónico institucional o utilizando las herramientas proporcionadas por el Campus Virtual. Los estudiantes en 5ª, 6ª y 7ª convocatoria tendrán prioridad frente al resto de compañeros en el horario de tutoría del profesor.

Atención presencial a grupos de trabajo

Se usará para aclarar dudas, orientar las actividades colectivas dirigidas, mejorar el rendimiento académico, hacer un seguimiento del trabajo realizado y modelar las habilidades sociales. El profesor atenderá consultas del alumnado en su horario de tutorías. Las citas se pueden concertar a través del correo electrónico institucional o utilizando las herramientas proporcionadas por el Campus Virtual.

Atención telefónica

El profesor de la asignatura atenderá consultas telefónicas del alumnado en su horario de tutorías.

Atención virtual (on-line)

El profesor de la asignatura estará disponible para atender a los estudiantes virtualmente a través del correo electrónico institucional o utilizando las herramientas proporcionadas por el Campus Virtual.

Datos identificativos del profesorado que la imparte.

Datos identificativos del profesorado que la imparte

Dr./Dra. José María Quinteiro González (COORDINADOR)

Departamento: 238 - INGENIERÍA TELEMÁTICA

Ámbito: 560 - Ingeniería Telemática

Área: 560 - Ingeniería Telemática

Despacho: INGENIERÍA TELEMÁTICA

Teléfono: 928451236 **Correo Electrónico:** josemaria.quinteiro@ulpgc.es

Dr./Dra. Francisco Javier Miranda González

Departamento: 238 - INGENIERÍA TELEMÁTICA

Ámbito: 560 - Ingeniería Telemática

Área: 560 - Ingeniería Telemática

Despacho: INGENIERÍA TELEMÁTICA

Teléfono: 928451240 **Correo Electrónico:** javier.miranda@ulpgc.es

D/Dña. José Évora Gómez

Departamento: 238 - INGENIERÍA TELEMÁTICA

Ámbito: 560 - Ingeniería Telemática

Área: 560 - Ingeniería Telemática

Despacho: INGENIERÍA TELEMÁTICA

Teléfono: **Correo Electrónico:** jose.evora@ulpgc.es

Dr./Dra. José Miguel Santana Núñez

Departamento: 238 - INGENIERÍA TELEMÁTICA

Ámbito: 560 - Ingeniería Telemática

Área: 560 - Ingeniería Telemática

Despacho: INGENIERÍA TELEMÁTICA

Teléfono: **Correo Electrónico:** josemiguel.santana@ulpgc.es

Bibliografía

[1 Básico] Introduction to Algorithms /

Thomas H. Cormen ... [et al.].

The Mit Press., Cambridge (Massachusetts) ; (2009) - (3rd ed.)

978-0-262-03384-8

[2 Básico] Algorithms Illuminated (Part 1)

Tim Roughgarden

Soundlikeyourself Publishing, LLC.

978-0-9992829-0-8

[3 Básico] Algorithms Illuminated (Part 2)

Tim Roughgarden

Soundlikeyourself Publishing, LLC.

978-9992829-2-2

[4 Básico] Algorithms Illuminated (Part 3)

Tim Roughgarden

Soundlikeyourself Publishing, LLC.

978-0-9992829-4-6

[5 Recomendado] The design of approximation algorithms /

David P. Williamson, David B. Shmoys.

Cambridge University Press., Cambridge [etc.] : (2011)

978-0-521-19527-0 (*Observaciones: Tema 5*)

[6 Recomendado] Search Methodologies [Introductory Tutorials in Optimization and Decision Support Techniques /

edited by Edmund K. Burke, Graham Kendall.

Springer Science+Business Media, LLC., Boston, MA : (2005)

9780387283562 (*Observaciones: Tema 6*)

[7 Recomendado] Computational Combinatorial Optimization [Optimal or Provably Near-Optimal Solutions /

edited by Michael Jünger, Denis Naddef.

Springer-Verlag Berlin Heidelberg., Berlin, Heidelberg : (2001)

9783540455868 (*Observaciones: Tema 7*)

[8 Recomendado] Computers and intractability: a guide to the theory of NP-completeness.

Garey, Michael R.

Freeman., New York : (1979)

0716710455 (*Observaciones: Tema 3*)

[9 Recomendado] Model building in mathematical programming /

H. Paul Williams.

John Wiley & Sons., Chichester [etc.] : (2003) - (4th ed. reprinted.)

0471997889 (*Observaciones: Tema 4*)

[10 Recomendado] Introducción al diseño y análisis de algoritmos: un enfoque estratégico /

R.C.T. Lee ... [et al.] ; revisión técnica: Miguel A. Orozco Malo, Jorge Valeriano Assem, Carlos Villegas Quezada.

McGraw-Hill., México, D.F : (2007)

978-970-10-6124-4
