26-27

GRADO EN INGENIERÍA EN INTELIGENCIA ARTIFICIAL
SEGUNDO CURSO

GUÍA DE ESTUDIO PÚBLICA

ALGORITMIA PARA LA IA

CÓDIGO 71032050

UNED

26-27

ALGORITMIA PARA LA IA

CÓDIGO 71032050

ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA
EQUIPO DOCENTE
HORARIO DE ATENCIÓN AL ESTUDIANTE
TUTORIZACIÓN EN CENTROS ASOCIADOS
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE
RESULTADOS DE APRENDIZAJE
CONTENIDOS
METODOLOGÍA
SISTEMA DE EVALUACIÓN
BIBLIOGRAFÍA BÁSICA
BIBLIOGRAFÍA COMPLEMENTARIA
RECURSOS DE APOYO Y WEBGRAFÍA
IGUALDAD DE GÉNERO

NOMBRE DE LA ASIGNATURA	ALGORITMIA PARA LA IA
CÓDIGO	71032050
CURSO ACADÉMICO	2026/2027
DEPARTAMENTO	LENGUAJES Y SISTEMAS INFORMÁTICOS
TÍTULO EN QUE SE IMPARTE	GRADO EN INGENIERÍA EN INTELIGENCIA ARTIFICIAL
CURSO	SEGUNDO CURSO
PERIODO	SEMESTRE 2
Nº ETCS	6
HORAS	150.0
IDIOMAS EN QUE SE IMPARTE	CASTELLANO

PRESENTACIÓN Y CONTEXTUALIZACIÓN

El objetivo de esta asignatura, perteneciente a la materia **Programación y Algoritmia**, es que el estudiante adquiera los fundamentos de la algoritmia aplicada a la Inteligencia Artificial, desarrollando la capacidad de diseñar, analizar e implementar algoritmos eficientes para la resolución de problemas complejos. Para ello, se abordan tanto los principios de diseño algorítmico como su aplicación práctica en distintos contextos.

A lo largo del curso se estudian los principales esquemas algorítmicos (voraces, divide y vencerás, programación dinámica, vuelta atrás y ramificación y poda), junto con el análisis de la complejidad computacional, proporcionando herramientas para evaluar la eficiencia y viabilidad de las soluciones. Asimismo, se introducen algoritmos de búsqueda fundamentales en IA, que permiten modelar y resolver problemas en espacios de estados, incluyendo técnicas de búsqueda heurística y exploración informada.

La asignatura se apoya en los conocimientos adquiridos en **Fundamentos de Programación** y **Programación Orientada a Objetos**, donde el estudiante desarrolla las competencias básicas de programación y abstracción, y mantiene una estrecha relación con **Estructuras de Datos y Algoritmos**, en la que se introducen las estructuras y técnicas que aquí se amplían y aplican a problemas más complejos. En continuidad, **Algoritmia para la IA** profundiza en estos contenidos, orientándolos específicamente hacia la resolución de problemas en Inteligencia Artificial, incorporando técnicas avanzadas de búsqueda, heurísticas y optimización.

En este contexto, la algoritmia constituye una disciplina esencial del pensamiento computacional y resulta especialmente relevante en Inteligencia Artificial, donde la eficiencia, la optimización y la capacidad de explorar grandes espacios de soluciones son aspectos clave. La asignatura proporciona así las bases necesarias para comprender y desarrollar algoritmos aplicados en áreas como la planificación, la optimización o el aprendizaje automático.

REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA

Para cursar esta asignatura, es necesario haber superado las siguientes asignaturas del grado:

•Fundamentos de programación

•Programación orientada a objetos

•Estructuras de datos y algoritmos

Específicamente, se requieren los siguientes conocimientos:

•Conocimientos básicos de programación, impartidos en la asignatura de Fundamentos de Programación

•Programación orientada a objetos y Python, impartidos en la asignatura de Programación Orientada a Objetos

•Estructuras de datos, impartidas en la asignatura de Estructuras de Datos y Algoritmos

•Nociones de lógica y matemática discreta que se imparten en la asignatura de Lógica y Estructuras Discretas

•Nociones de álgebra que se imparten en la asignatura de Fundamentos Algebraicos para la IA

EQUIPO DOCENTE

Nombre y Apellidos	LAURA PLAZA MORALES (Coordinador/a de asignatura)
Correo Electrónico	lplaza@lsi.uned.es
Teléfono	91398-8919
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS

Nombre y Apellidos	M LOURDES ARAUJO SERNA
Correo Electrónico	lurdes@lsi.uned.es
Teléfono	91398-7318
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	LENGUAJES Y SISTEMAS INFORMÁTICOS

HORARIO DE ATENCIÓN AL ESTUDIANTE

El alumno dispone de varios mecanismos para facilitar su aprendizaje:

•Tutores en los centros asociados. Serán los encargados de desarrollar las sesiones presenciales de seguimiento y control de las prácticas. Para realizarlas, el alumno debe ponerse en contacto con su tutor al comienzo del semestre, para conocer cuanto antes los horarios de las sesiones previstas.

•Tutorías presenciales o virtuales en el centro asociado correspondiente.

•Entorno virtual de la asignatura (accesible desde el portal de la UNED, www.uned.es). Ese entorno será el mecanismo básico de comunicación del equipo docente con estudiantes y

tutores. Dispone además de foros donde los alumnos podrán plantear sus dudas para que sean respondidas por los tutores o por el propio equipo docente.

TUTORÍAS: El equipo docente atenderá dudas, preferentemente, a través de los foros del entorno vitual. Para consultas de carácter personal, pueden dirigirse al equipo docente a través del correo **algoritmia@lsi.uned.es**.

El horario de guardia es el siguiente horario (días lectivos):

•Lourdes Araujo Serna: Jueves de 10:00 a 14:00 horas. 91398-7318

•Laura Plaza Morales (Coordinadora de la asignatura): Jueves de 10:00 a 14:00 horas. 91398-8919

La atención presencial se realiza en la ETSI de Informática de la UNED en C/Juan del Rosal, 16 28040 Madrid.

El Equipo Docente no resuelve problemas producidos a la hora de instalar, compilar o depurar los programas que vayan generando para el estudio de la asignatura o la práctica. Las dudas al respecto deben plantearse al Tutor asignado por el Centro Asociado.

TUTORIZACIÓN EN CENTROS ASOCIADOS

Las sesiones de tutoría se imparten presencialmente y en línea desde el Centro Asociado de referencia o, en su defecto, desde otro Centro Asociado. A continuación se muestran los centros asociados y extensiones en las que se imparten tutorías de la asignatura.

COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Competencias generales:

Competencias cognitivas superiores: selección y manejo adecuado de conocimientos, recursos y estrategias cognitivas de nivel superior apropiados para el afrontamiento y resolución de diversos tipos dtareas/problemas con distinto nivel de complejidad y novedad: Análisis y Síntesis. Aplicación de los conocimientos a la práctica Resolución de problemas en entornos nuevos o poco conocidos. Pensamiento creativo. Razonamiento crítico. Toma de decisiones.

Competencias en el uso de las herramientas y recursos de la Sociedad del Conocimiento: Manejo de las TIC. Competencia en la búsqueda de información relevante. Competencia en la gestión y organización de la información. Competencia en la recolección de datos, el manejo de bases de datos y su presentación.

Competencias específicas:

Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para el tratamiento automático de la información por medio de sistemas computacionales y para la resolución de problemas propios de la ingeniería.

Capacidad para analizar empíricamente el comportamiento de algoritmos mediante experimentación, comparando su rendimiento en términos de tiempo, espacio y calidad de solución.

Capacidad para abstraer y formalizar problemas del mundo real en términos algorítmicos, identificando las estructuras de datos y estrategias de resolución más adecuadas.
Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
Capacidad para modelar problemas como espacios de estados y aplicar algoritmos de búsqueda, tanto no informada como informada, utilizando funciones heurísticas para mejorar la eficiencia en la resolución.

RESULTADOS DE APRENDIZAJE

- **RA06 -** Al finalizar el aprendizaje, el o la estudiante será capaz de analizar las necesidades en algorítmica, complejidad computacional, programación, sistemas operativos, bases de datos, estructura, interconexión de sistemas informáticos necesarios para la resolución de problemas de ciencias e ingeniería, de acuerdo con los principios de calidad, fiabilidad y seguridad necesarios. TIPO: Habilidades o destrezas.
- **RA08 -** Al finalizar el aprendizaje, el o la estudiante será capaz de demostrar conocimiento de los procedimientos algorítmicos básicos y los tipos de datos de las tecnologías informáticas necesarios para la resolución de los problemas de inteligencia artificial. TIPO: Conocimientos o contenidos.
- **RA09 -** Al finalizar el aprendizaje, el o la estudiante será capaz de identificar los algoritmos, estructuras de datos, paradigmas de la programación y lenguajes más adecuados para asegurar la fiabilidad, seguridad y calidad de aplicaciones en problemas que requieran una solución de inteligencia artificial. TIPO: Habilidades o destrezas.

CONTENIDOS

Tema 1: Introducción

Este tema introduce el concepto de algoritmo como base para la resolución de problemas, destacando su papel clave en la Inteligencia Artificial. Se enfatiza la importancia de la eficiencia, no solo en la corrección de las soluciones, sino en su coste en tiempo y recursos, lo que conduce al estudio de la complejidad computacional. Se presentan los principales paradigmas algorítmicos y su aplicación en IA, especialmente en problemas de búsqueda, optimización y aprendizaje, donde a menudo se requieren heurísticas. Asimismo, se contextualiza la evolución histórica de la algoritmia y su convergencia con la IA. Finalmente, se ofrece una visión estructurada del contenido, organizada en análisis de algoritmos, esquemas algorítmicos y algoritmos de búsqueda en IA.

Tema 2: Análisis de algoritmos

Este tema introduce el análisis de algoritmos como herramienta para evaluar no solo si una solución es correcta, sino también si es eficiente y escalable. Se distinguen el análisis empírico, basado en medir ejecuciones reales, y el análisis teórico, centrado en estimar el coste de forma independiente del hardware o del lenguaje. Se estudian la complejidad temporal y espacial, los casos mejor, medio y peor, y la notación asintótica. Además, se presentan reglas básicas para calcular el número de operaciones en instrucciones, bucles, condicionales y funciones recursivas. Finalmente, se introducen las recurrencias por sustracción y por división, fundamentales para analizar algoritmos recursivos y esquemas como divide y vencerás.

Tema 3: Algoritmos Voraces

El principal objetivo de este tema es presentar el esquema voraz y los tipos de problemas más representativos que es capaz de resolver.
El esquema voraz (greedy algorithms) se aplica a problemas de optimización en los que la solución se puede construir paso a paso sin necesidad de reconsiderar decisiones ya tomadas. En cada etapa del proceso, se selecciona la opción que parece más prometedora en ese momento según un criterio de selección local, sin tener en cuenta el efecto global de dicha elección. Esta estrategia resulta eficiente en términos computacionales, ya que evita la exploración exhaustiva del espacio de soluciones. Sin embargo, no garantiza siempre una solución óptima global. Por ello su aplicabilidad requiere demostrar que el algoritmo propuesto lleva efectivamente a la solución óptima (demostración de optimalidad).

Tema 4: Divide y vencerás

Este tema describe la estrategia de Divide y Vencerás, que es una técnica algorítmica que se basa en la descomposición de un problema en subproblemas de su mismo tipo, lo que permite disminuir la complejidad y en algunos casos, paralelizar la resolución de los mismos.

Tema 5: Programación dinámica

Este tema describe la técnica de Programación Dinámica, que es una estrategia algorítmica que se basa en la resolución de problemas mediante la descomposición en subproblemas solapados, cuyas soluciones se almacenan y reutilizan, lo que permite evitar cálculos repetidos y reducir significativamente el coste computacional, a costa de un mayor uso de memoria y, en muchos casos, facilitando la obtención de soluciones óptimas en problemas de optimización.

Tema 6: Vuelta atrás

En este tema se presenta el esquema de Vuelta Atrás o retroceso. Este esquema se aplica a problemas en los que sólo podemos recurrir a una búsqueda exhaustiva, recorriendo el espacio de todas las posibles soluciones hasta que encontremos una de ellas o hasta que hayamos explorado todas las opciones, concluyendo así que no existe la solución buscada.

Tema 7: Ramificación y poda

Este tema describe el esquema de ramificación y poda (*branch and bound*) que se emplea en aquellos problemas en los que el objetivo es la optimización de uno o más criterios en la solución alcanzada. A diferencia de la búsqueda exhaustiva, este esquema evita explorar ramas que no puedan conducir a una solución mejor que la ya conocida, aplicando un mecanismo de poda basado en cotas (*bounds*) que descartan subárboles enteros cuando se detecta que no pueden mejorar la solución actual. De este modo, se reduce drásticamente el número de configuraciones evaluadas sin renunciar a la garantía de optimalidad.

Tema 8: Algoritmos de búsqueda en Inteligencia Artificial

En este tema se estudian los algoritmos de búsqueda como base de la Inteligencia Artificial, entendiendo cómo resolver problemas mediante la exploración de un espacio de estados. Se introduce la representación formal de los problemas y su modelado como grafos o árboles. Se analizan las estrategias de búsqueda no informada, como la búsqueda en profundidad, en anchura y de coste uniforme. También se estudian las estrategias informadas basadas en heurísticas, destacando el algoritmo A*. Además, se aborda la búsqueda con adversarios mediante minimax y poda alfa-beta. Finalmente, se presentan criterios de evaluación como completitud, optimalidad y complejidad.

METODOLOGÍA

La metodología de estudio es la propia de la enseñanza a distancia en la UNED, que combina el trabajo autónomo del estudiante con los materiales disponibles en el curso virtual y la guía de estudio, con la tutorización en su centro asociado, el asesoramiento del equipo docente y la interacción con tutores y compañeros de estudios en el campus virtual y en el centro asociado.
Cada estudiante trabajará con contenidos teórico-prácticos utilizando el material de estudio, la guía de estudio y el material complementario.
Se recomienda abordar cada tema siguiendo una secuencia progresiva de aprendizaje. En primer lugar, el estudiante debe realizar una lectura comprensiva del material teórico, prestando especial atención a los conceptos fundamentales, los esquemas algorítmicos y los ejemplos ilustrativos. A continuación, es conveniente resolver los ejercicios de práctica

propuestos en el tema para afianzar la comprensión de los contenidos. Una vez adquirida una visión general, se recomienda estudiar y reproducir de forma autónoma los ejercicios resueltos, intentando comprender el razonamiento seguido en cada paso y reconstruyendo las soluciones sin consultar inicialmente la resolución. Posteriormente, el estudiante debería enfrentarse por sí mismo a los ejercicios recomendados, aplicando los conocimientos adquiridos y desarrollando su capacidad de diseño y análisis de algoritmos. Finalmente, para profundizar en los contenidos y consolidar las competencias de resolución de problemas, se aconseja consultar la bibliografía complementaria y utilizarla como fuente de nuevos ejercicios y casos de estudio que permitan ampliar la experiencia práctica más allá del material básico de la asignatura.

Se realizarán **dos prácticas** bajo la supervisión del tutor en el centro asociado, con las herramientas y directrices preparadas por el equipo docente. La participación en las **sesiones presenciales de prácticas organizadas por el tutor en el centro asociado** es un requisito obligado para superar las prácticas, así como su correcta defensa ante el tutor.

Aunque estamos en la era de la **Inteligencia Artificial Generativa (IAG)**, las prácticas tienen como objetivo ayudar al estudiante a asimilar los contenidos teóricos y, por tanto, constituyen una parte fundamental de la asignatura. Por ello, no está permitido el uso de herramientas de IAG para la realización de los trabajos prácticos.

Para aprobar la asignatura es requisito imprescindible entregar y superar los trabajos prácticos (ver "sistema de evaluación").

SISTEMA DE EVALUACIÓN

TIPO DE PRUEBA PRESENCIAL

Tipo de examen	Examen mixto
Preguntas test	6
Preguntas desarrollo	1
Duración del examen	120 (minutos)

Material permitido en el examen

Ninguno.

Criterios de evaluación

La nota del examen representa el 80% de la valoración final de la asignatura (el 20% restante corresponde a las prácticas).
El examen consta de 6 cuestiones tipo test y un ejercicio práctico.
Cada cuestión contestada correctamente vale 1 punto.
Cada cuestión contestada incorrectamente baja la nota en 0.3 puntos.
Debe obtenerse un mínimo de 3 puntos en las cuestiones para que el problema sea valorado (con 3 cuestiones correctas y alguna incorrecta el examen está suspenso).
El ejercicio práctico consiste en la resolución de un problema mediante la aplicación de alguno de los esquemas algorítmicos estudiados en la asignatura. Se valorará la justificación de las decisiones de diseño adoptadas, la calidad técnica de la solución desarrollada y, cuando proceda, el análisis de coste computacional del algoritmo propuesto. Se valorará también la claridad de la presentación y la exposición.
La nota total del examen debe ser al menos de 4.5 para aprobar.

% del examen sobre la nota final	80
Nota del examen para aprobar sin PEC	
Nota máxima que aporta el examen a la calificación final sin PEC	
Nota mínima en el examen para sumar la PEC	
Comentarios y observaciones	

PRUEBAS DE EVALUACIÓN CONTINUA (PEC)

¿Hay PEC?	No

Descripción

Esta asignatura tiene Prácticas en lugar de PED.

Criterios de evaluación

Ponderación de la PEC en la nota final

Fecha aproximada de entrega	Las fechas de entrega las fijan los tutores de los Centros Asociados.
Comentarios y observaciones	

OTRAS ACTIVIDADES EVALUABLES

¿Hay otra/s actividad/es evaluable/s?	Si

Descripción

El trabajo del curso incluye la realización de dos prácticas obligatorias de programación en lenguaje Python. El objetivo de estas prácticas es ayudar al alumno a la comprensión de los temas tratados, así como hacerle ver su aplicación.
El enunciado de las prácticas estará disponible en el curso virtual de la asignatura.

Criterios de evaluación

Las prácticas son corregidas por los Tutores de los Centros Asociados. La nota asignada por el tutor podrá incrementar hasta un máximo de 2 puntos (1 por cada práctica) en la nota final de la asignatura de acuerdo con el siguiente baremo:

Sobresaliente -> + 1

Notable -> + 0.75

Aprobado -> + 0.5

Ponderación en la nota final	20
Fecha aproximada de entrega	
Comentarios y observaciones	

Las prácticas tienen sesiones de asistencia obligatoria y se organizan desde los centros asociados bajo la responsabilidad de cada tutor, por lo que los alumnos deben ponerse en contacto con ellos lo antes posible al comienzo el curso para conocer:

El calendario de entrega de prácticas, tanto para la convocatoria de febrero como para la de septiembre, y de las sesiones presenciales obligatorias de las prácticas.

La forma de entrega.

Los programas deben compilar independientemente del entorno de edición que haya sido usado, debiendo el alumno especificar claramente los pasos a seguir para el correcto funcionamiento de las aplicaciones generadas en las prácticas. De no cumplirse este requisito y en caso de no poder ejecutar una práctica, ésta se considerará suspensa. Cualquier copia en las prácticas dará lugar a un suspenso para todo el curso académico.

¿CÓMO SE OBTIENE LA NOTA FINAL?

El examen solo puntúa si las prácticas están aprobadas.

La nota de un examen no se guarda para siguientes convocatorias. La de las prácticas se guarda dentro del mismo Curso Académico y también, si ambas prácticas están aprobadas, se guarda como máximo de un curso para otro.

La nota del examen representa el 80% de la valoración final de la asignatura y las prácticas el 20% restante.

BIBLIOGRAFÍA BÁSICA

El texto base estará disponible en el curso virtual:

Algoritmia para la Inteligencia Artificial

Lourdes Araujo Serna, Raquel Martínez Unanue, Laura Plaza Morales, Miguel Rodríguez Artacho.

BIBLIOGRAFÍA COMPLEMENTARIA

ISBN(13):9788420538495
Título:ESTRUCTURAS DE DATOS Y MÉTODOS ALGORÍTMICOS: EJERCICIOS RESUELTOS1ª
Autor/es:Martí Oliet, Narciso ; Verdejo López, José Alberto ; Ortega Mallén, Yolanda ;
Editorial:PEARSON ALHAMBRA

ISBN(13):9788436236224
Título:ESQUEMAS ALGORÍTMICOS. ENFOQUE METODOLÓGICO Y PROBLEMAS RESUELTOS1ª
Autor/es:Gonzalo Arroyo, Julio ; Rodríguez Artacho, Miguel ;
Editorial:U.N.E.D.

ISBN(13):9788489660007
Título:FUNDAMENTOS DE ALGORITMIA1ª
Autor/es:Bratley, Paul ; Brassard, Gilles ;
Editorial:PRENTICE-HALL

ISBN(13):9788420540030
Título:INTELIGENCIA ARTIFICIAL: UN ENFOQUE MODERNO2ª
Autor/es:Norvig, Peter ; Russell, Stuart ;
Editorial:PRENTICE-HALL

ISBN(13):9788480040907
Título:INTELIGENCIA ARTIFICIAL :1 ed., 1 reimp
Autor/es:Borrajo Millán, Daniel ;
Editorial:CENTRO DE ESTUDIOS RAMÓN ARECES

ISBN(13):9788420541914
Título:DISEÑO DE PROGRAMAS. FORMALISMO Y ABSTRACCIÓNnull
Autor/es:Peña Marí, Ricardo ;
Editorial:PRENTICE-HALL

Técnicas de Diseño de Algoritmos.
GUEREQUETA, R. y VALLECILLO, A.:
Ed. Universidad de Málaga (1998).
https://www.lcc.uma.es/~av/Libro/
En estos libros pueden encontrarse ejemplos de uso de las estructuras de datos y los esquemas algoritmicos propuestos distintos a los presentados en el texto base.

RECURSOS DE APOYO Y WEBGRAFÍA

Además de esta guía del curso y la guía de estudio (que detalla el plan de trabajo propuesto para el alumno), el equipo docente proporcionará material adicional a través del entorno de enseñanza virtual de la asignatura.

IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.