Máster Título Propio

Inteligencia Artificial e Ingeniería del Conocimiento



tech universidad tecnológica



tech universidad tecnológica

Máster Título Propio

Inteligencia Artificial e Ingeniería del Conocimiento

Modalidad: **Online**
Duración: **12 meses**
Titulación: **TECH Universidad Tecnológica**
Horas lectivas: **1.500 h.**
Acceso web: **www.techtitute.com/informatica/master/master-inteligencia-artificial-ingenieria-conocimiento**

Índice

01 Presentación — pág. 4

02 Objetivos — pág. 8

03 Competencias — pág. 14

04 Estructura y contenido — pág. 18

05 Metodología — pág. 30

06 Titulación — pág. 38

01

Presentación

El programa en Este programa está diseñado para que los profesionales del ámbito de la Ingeniería se sumerjan en el apasionante mundo de la Inteligencia Artificial y la Ingeniería del Conocimiento. A través de una capacitación de alta competencia podrá dar un paso sólido y solvente en este terreno, consiguiendo las habilidades personales y profesionales necesarias para ejercer como un experto en la materia. Un programa completo y eficaz que le impulsará al más alto nivel de competencia.





Conviértete en uno de los profesionales más demandados del momento. Capacítate como Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento con este completísimo programa"

Los desarrollos basados en la Inteligencia Artificial han llegado a numerosas aplicaciones dentro del campo de la Ingeniería. Desde la automatización de numerosos procedimientos en la Industria y las empresas, hasta el propio Control de Procesos. Esto hace que los profesionales de la ingeniería necesiten conocer y dominar el funcionamiento de estas complejas técnicas.

Este conocimiento imprescindible se convierte además, en el primer paso para acceder a la propia capacidad de desarrollo de este tipo de tecnología.

A lo largo de esta capacitación se ofrece un panorama real de trabajo para poder valorar la conveniencia de su aplicación en proyecto propio, valorando sus indicaciones reales, su manera de desarrollo y las expectativas que se pueden tener con respecto a los resultados.

Mediante la experiencia conseguirá aprender cómo se desarrollan los conocimientos necesarios para avanzar en esta área de trabajo. Dicho aprendizaje, que necesita obligatoriamente de la experiencia, se concilia a través del teleaprendizaje y enseñanza práctica ofreciendo una opción única para dar a su CV el impulso que busca..

Incorpórate a la élite, con esta capacitación de alta eficacia instructiva y abre nuevos caminos a tu progreso profesional"

Éste **Master Título en Inteligencia Artificial e Ingeniería del Conocimiento**, contiene el programa más completo y actualizado del mercado. Sus características más destacadas son:

- Última tecnología en software de enseñanza online
- Sistema docente intensamente visual, apoyado en contenidos gráficos y esquemáticos de fácil asimilación y comprensión
- Desarrollo de casos prácticos presentados por expertos en activo
- Sistemas de vídeo interactivo de última generación
- Enseñanza apoyada en la telepráctica
- Sistemas de actualización y reciclaje permanente
- Aprendizaje autoregulable: total compatibilidad con otras ocupaciones
- Ejercicios prácticos de autoevaluación y constatación de aprendizaje
- Grupos de apoyo y sinergias educativas: preguntas al experto, foros de discusión y conocimiento
- Comunicación con el docente y trabajos de reflexión individual
- Disponibilidad de los contenidos desde cualquier dispositivo fijo o portátil con conexión a internet
- Bancos de documentación complementaria disponible permanentemente, incluso después del programa

Un Máster Título Propio que te capacitará para trabajar en todos los ámbitos de la Inteligencia Artificial e Ingeniería del Conocimiento con la solvencia de un profesional de alto nivel"

El personal docente está integrado por profesionales de diferentes ámbitos relacionados con esta especialidad. De esta manera se ofrece el objetivo de actualización instructiva que que se pretende. Un cuadro multidisciplinar de profesionales capacitados y experimentados en diferentes entornos, que desarrollarán los conocimientos teóricos de manera eficiente, pero sobre todo, pondrán a tu servicio los conocimientos prácticos derivados de su propia experiencia: una de las cualidades diferenciales de esta capacitación.

Este dominio de la materia se complementa con la eficacia del diseño metodológico. Elaborado por un equipo multidisciplinario de expertos en e-learning, el método integra los últimos avances en tecnología educativa. De esta manera, podrá estudiar con un elenco de herramientas multimedia cómodas y versátiles que le darán la operatividad que necesita en su capacitación.

El diseño de este programa toma como referencia el Aprendizaje Basado en Problemas: un planteamiento que concibe el aprendizaje como un proceso eminentemente práctico. Para conseguirlo de forma remota se hace uso de la telepráctica. Con la ayuda de un novedoso sistema de vídeo interactivo y el Learning From an Expert podrá adquirir los conocimientos como si estuviese enfrentándose al supuesto que está aprendiendo en el momento. Un concepto que permitirá integrar y fijar el aprendizaje de una manera más realista y permanente.

Con un diseño metodológico que se apoya en técnicas de enseñanza contrastadas por su eficacia, este novedoso Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento te llevará a través de diferentes abordajes docentes para permitirte aprender de forma dinámica y eficaz.

Nuestro innovador concepto de telepráctica te dará la oportunidad de aprender mediante una experiencia inmersiva, que te proporcionará una integración más rápida y una visión mucho más realista de los contenidos: "learning from an expert".



02

Objetivos

El objetivo es capacitar profesionales altamente cualificados para la experiencia laboral. Un objetivo que se complementa, de manera global, con el impulso de un desarrollo humano que siente las bases de una sociedad mejor. Este objetivo se materializa ayudando a los profesionales a acceder a un nivel de competencia y control mucho mayor. Una meta que, en tan apenas unos meses podrá dar por adquirida con una capacitación de alta intensidad y precisión.



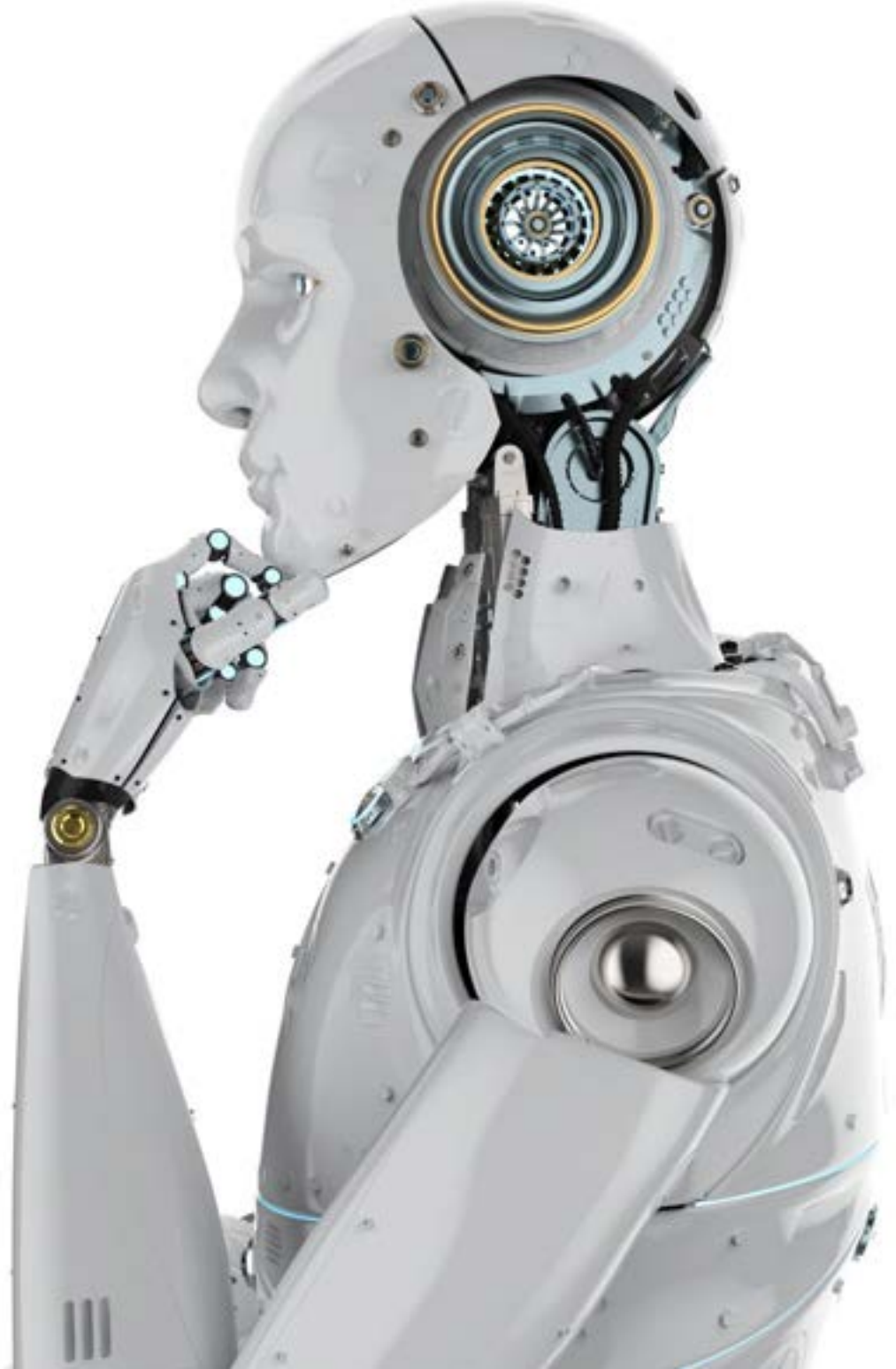


Si tu objetivo es reorientar tu capacidad hacia nuevos caminos de éxito y desarrollo, este es tu programa: una capacitación que aspira a la excelencia"

Objetivos generales

- Formar científica y tecnológicamente para el ejercicio de la ingeniería informática
- Obtener conocimientos amplios en el campo de la computación
- Obtener conocimientos amplios en el campo de la estructura de computadoras
- Adquirir los conocimientos necesarios en ingeniería del software
- Revisar las bases matemáticas, estadísticas y físicas imprescindibles para esta materia

No pierdas la oportunidad y ponte al día en las novedades en el uso de antihemorrágicos para incorporarlos a tu práctica médica diaria"



Objetivos específicos

Módulo 1. Fundamentos de programación

- Comprender la estructura básica de un ordenador, el software y de los lenguajes de programación de propósito general
- Aprender a diseñar e interpretar algoritmos, que son la base necesaria para poder desarrollar programas informáticos
- Entender los elementos esenciales de un programa informático, como son los distintos tipos de datos, operadores, expresiones, sentencias, E/S y sentencias de control
- Comprender los distintas estructuras de datos disponibles en los lenguajes de programación de propósito general tanto estáticas como dinámicas, así como adquirir los conocimientos esenciales para el manejo de ficheros
- Conocer las distintas técnicas de pruebas en los programas informáticos y la importancia de generar una buena documentación junto con un buen código fuente
- Aprender los conceptos básicos del lenguaje de programación C++, uno de los más usados a nivel mundial

Módulo 2. Estructura de datos

- Aprender los fundamentos de la programación en el lenguaje C++, incluyendo clases, variables, expresiones condicionales y objetos
- Entender los tipos abstractos de datos, los tipos de estructuras de datos lineales, estructuras de datos jerárquicas simples y complejas. así como su implementación en C++
- Comprender el funcionamiento de estructuras de datos avanzadas distintas de las habituales
- Conocer la teoría y la práctica relacionada con el uso de montículos y colas de prioridad
- Aprender el funcionamiento de las tablas hash, como tipos abstractos de datos y funciones
- Entender la teoría de grafos, así como algoritmos y concepto avanzados sobre grafos

Módulo 3. Algoritmia y complejidad

- Aprender las principales estrategias de diseño de algoritmos, así como los distintos métodos y medidas para de cálculo de los mismos
- Conocer los principales algoritmos de ordenación usados en el desarrollo de software
- Entender el funcionamiento de los distintos algoritmos con árboles, *Heaps* y grafos
- Comprender el funcionamiento de los algoritmos *Greedy*, su estrategia y ejemplos de su uso en los principales problemas conocidos. Conoceremos también el uso de algoritmos *Greedy* sobre grafos
- Aprenderemos las principales estrategias de búsqueda de caminos mínimos, con el planteamiento de problemas esenciales del ámbito y algoritmos para su resolución
- Entender la técnica de backtracking y sus principales usos, así como otras técnicas alternativas

Módulo 4. Diseño avanzado de algoritmos

- Profundizar en el diseño avanzado de algoritmos, analizando algoritmos recursivos y tipo divide y conquista, así como realizando análisis amortizado
- Comprender los conceptos de programación dinámica y los algoritmos para problemas NP
- Entender el funcionamiento de la optimización combinatoria, así como los distintos algoritmos de aleatorización y algoritmos paralelos
- Conocer y comprender el funcionamiento de los distintos métodos de búsqueda local y con candidatos
- Aprender los mecanismos de verificación de formal de programas y de programas iterativos, incluyendo la lógica de primer orden y el sistema formal de *Hoare*
- Aprender el funcionamiento de algunos de los principales métodos numéricos como el método de la bisección, el método de Newton Raphson y el método de la secante

Módulo 5. Lógica computacional

- Aprender los fundamentos de la lógica computacional, para qué sirve y su justificación de uso
- Conocer las distintas estrategias de formalización y deducción en la lógica proposicional, incluyendo el razonamiento natural, la deducción axiomática y la natural, así como las reglas primitivas del cálculo proposicional
- Adquirir los conocimientos avanzados en lógica proposicional, adentrándose en la semántica de la misma y en las principales aplicaciones de esta lógica como lo son los circuitos lógicos
- Entender la lógica de predicados tanto para el cálculo de deducción natural de predicados, como para las estrategias de formalización y deducción para lógica de predicados
- Entender las bases del lenguaje natural y de su mecanismo deductivo
- Introducir a la programación lógica usando el lenguaje PROLOG

Módulo 6. Inteligencia artificial e Ingeniería del Conocimiento

- Sentar las bases de la Inteligencia Artificial y la Ingeniería del Conocimiento, haciendo un breve recorrido por la historia de la Inteligencia Artificial hasta llegar a nuestros días
- Comprender los conceptos esenciales de la búsqueda en la Inteligencia Artificial, tanto de la búsqueda informada como de la no informada
- Entender el funcionamiento de la Inteligencia Artificial en juegos
- Aprender los conceptos fundamentales de las redes neuronales y el uso de los algoritmos genéticos
- Adquirir los mecanismos oportunos para representar el conocimiento, especialmente teniendo en cuenta la web semántica
- Comprender el funcionamiento de los sistemas expertos y los sistemas de soporte a la decisión

Módulo 7. Sistemas inteligentes

- Aprender todos los conceptos relacionados con la teoría de agentes y la arquitectura de agentes y su proceso de razonamiento
- Asimilar la teoría y la práctica detrás de los conceptos de información y conocimiento, así como las distintas maneras de representar el conocimiento
- Entender la teoría relacionada con las ontologías, así como aprender lenguajes para ontologías y software para la creación de ontologías
- Aprender distintos modelos de representación del conocimiento, como son vocabularios, taxonomías, tesauros y mapas mentales entre otros
- Comprender el funcionamiento de los razonadores semánticos, los sistemas basados en conocimiento y los sistemas expertos
- Conocer el funcionamiento de la web semántica, el estado actual y futuro de la misma, así como aplicaciones basadas en web semántica

Módulo 8. Aprendizaje automático y minería de datos

- Introducir los procesos de descubrimiento del conocimiento y conceptos básicos de aprendizaje automático
- Aprender los métodos de exploración y preprocesamiento de datos, así como distintos algoritmos basados en árboles de decisión
- Comprender el funcionamiento de los métodos bayesianos y los métodos de regresión y de respuesta continua
- Entender las distintas reglas de clasificación y la evaluación de clasificadores, para ello se aprenderá a usar matrices confusión y evaluación numérica, el estadístico Kappa y la curva ROC
- Adquirir conocimiento esenciales relativos a la minería de textos y procesamiento de lenguaje natural (NLP) y al *Clustering*
- Profundizar en el conocimiento de redes neuronales, desde redes neuronales simples hasta redes neuronales recurrentes



Módulo 9. Sistemas multiagente y percepción computacional

- Comprender los conceptos básicos y avanzados relacionados con agentes y sistemas multiagente
- Estudiar el estándar para agentes FIPA, teniendo en cuenta la comunicación entre agentes, la gestión de los mismos y la arquitectura entre otras cuestiones
- Profundizar en el aprendizaje de la plataforma JADE (*Java Agent DEvelopment Framework)* aprendiendo a programar en ella tanto conceptos básicos como avanzados, incluyendo temas de comunicación y descubrimiento de agentes
- Sentar las bases del procesamiento del lenguaje natural, como el reconocimiento automático del habla y la lingüística computacional
- Entender en profundidad en funcionamiento de la visión artificial, el análisis de imágenes digitales, la transformación y la segmentación de las mismas

Módulo 10. Computación bioinspirada

- Introducir el concepto de computación bioinspirada, así como comprender el funcionamiento de los distintos tipos de algoritmos de adaptación social y de algoritmos genéticos
- Profundizar en el estudio de los distintos modelos de computación evolutiva, conociendo sus estrategias, programación, algoritmos y modelos basados en estimación de distribuciones
- Entender las principales estrategias de exploración-explotación del espacio para algoritmos genéticos
- Comprender el funcionamiento de la programación evolutiva aplicada a problemas de aprendizaje y de los problemas multiobjetivo
- Aprender los conceptos esenciales relacionados con redes neuronales y entender el funcionamiento de casos de uso reales aplicados a áreas tan dispares como la investigación médica, la economía y la visión artificial

03

Competencias

Este Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento ha sido creado como herramienta de alta capacitación para el profesional. Su intensiva capacitación lo prepararán para poder trabajar en todos los ámbitos relacionados con la Inteligencia Artificial con la seguridad de un experto en la materia.



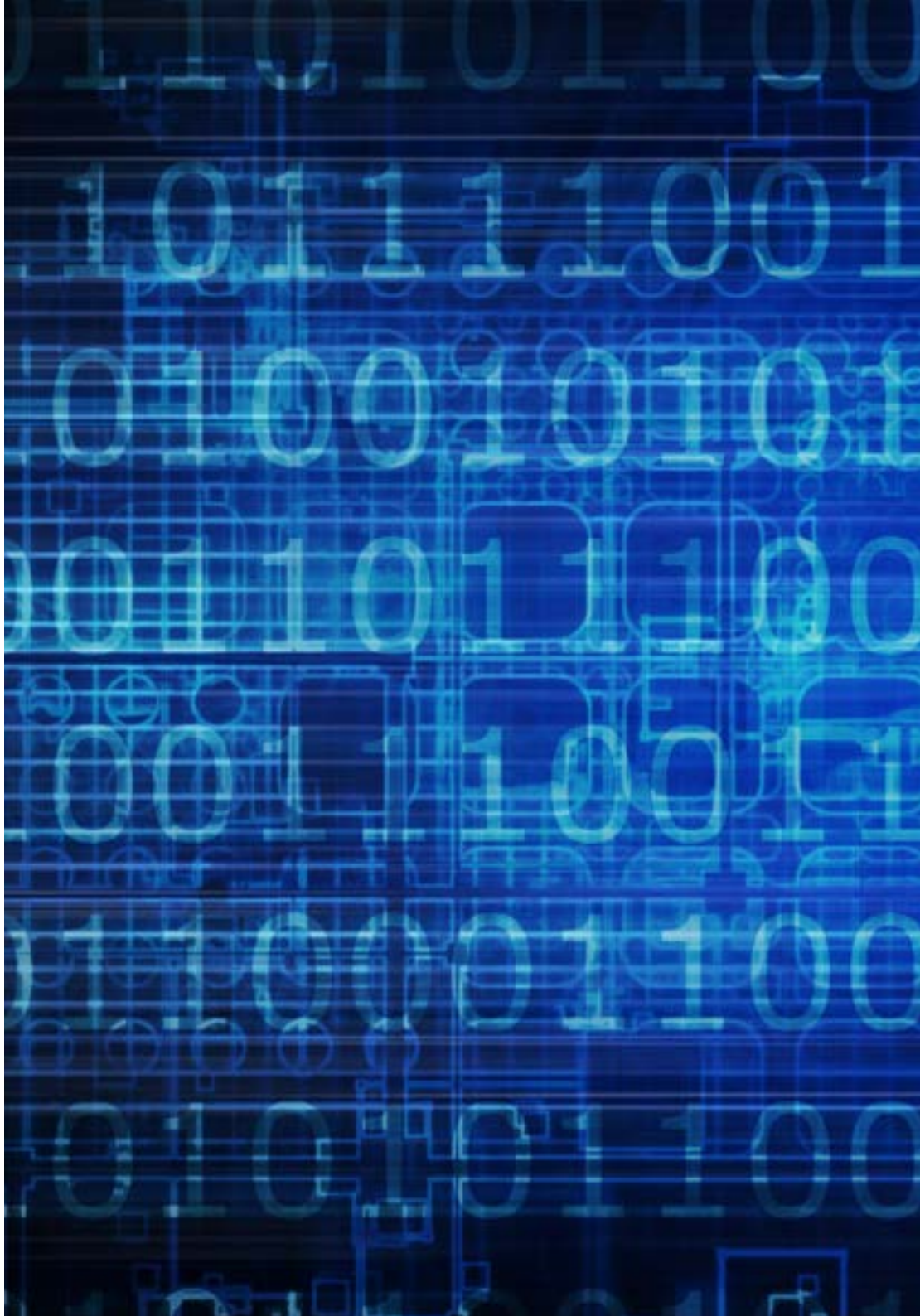


El Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento te proporcionará las competencias personales y profesionales imprescindibles para jugar un adecuado papel en cualquier situación profesional en este ámbito de intervención"

Competencia general

- Adquirir las habilidades necesarias para el ejercicio profesional de la ingeniería informática con el conocimiento de todos los factores necesarios para realizarlo con calidad y solvencia

Una experiencia de capacitación única, clave y decisiva para impulsar tu desarrollo profesional"





Competencias específicas

- Desarrollar una programación en el área de la inteligencia artifical teniendo en cuenta todos los factores de desarrollo de la misma
- Conocer con solvencia la estructura de datos en programación C++
- Diseñar algoritmos básicos y avanzados
- Entender la lógica computacional y aplicarla en el diseño de proyectos
- Saber acerca de Inteligencia Artificial, sus usos y sus desarrollos e implementar los propios proyectos
- Saber qué son, cómo funcionan y cómo se trabaja con sistemas inteligentes
- Dominar los conceptos básicos del aprendizaje automático
- Conocer JADE, FIPA, visión artificial y otros sistemas multiagentes
- Conocer los algoritmos de computación bioinspirada y las estrategias de utilización

04

Estructura y contenido

Los contenidos de este Máster Título Propio han sido desarrollados por diferentes expertos en la materia, con una finalidad clara: conseguir que el alumnado adquiera todas y cada una de las habilidades necesarias para convertirse en verdaderos expertos en todo lo relacionado con la Inteligencia Artificial.

Un programa completísimo y muy bien estructurado que le llevará hacia los más elevados estándares de calidad y éxito.





Un completísimo programa docente, estructurado en unidades didácticas muy bien desarrolladas, orientadas a un aprendizaje compatible con tu vida personal y profesional"

Módulo 1. Fundamentos de programación

1.1. Introducción a la programación
- 1.1.1. Estructura básica de un ordenador
- 1.1.2. Software
- 1.1.3. Lenguajes de programación
- 1.1.4. Ciclo de vida de una aplicación informática

1.2. Diseño de algoritmos
- 1.2.1. La resolución de problemas
- 1.2.2. Técnicas descriptivas
- 1.2.3. Elementos y estructura de un algoritmo

1.3. Elementos de un programa
- 1.3.1. Origen y características del lenguaje C++
- 1.3.2. El entorno de desarrollo
- 1.3.3. Concepto de programa
- 1.3.4. Tipos de datos fundamentales
- 1.3.5. Operadores
- 1.3.6. Expresiones
- 1.3.7. Sentencias
- 1.3.8. Entrada y salida de datos

1.4. Sentencias de control
- 1.4.1. Sentencias
- 1.4.2. Bifurcaciones
- 1.4.3. Bucles

1.5. Abstracción y Modularidad: funciones
- 1.5.1. Diseño modular
- 1.5.2. Concepto de función y utilidad
- 1.5.3. Definición de una función
- 1.5.4. Flujo de ejecución en la llamada de una función
- 1.5.5. Prototipo de una función
- 1.5.6. Devolución de resultados
- 1.5.7. Llamada a una función: parámetros
- 1.5.8. Paso de parámetros por referencia y por valor
- 1.5.9. Ámbito identificador

1.6. Estructuras de datos estáticas
- 1.6.1. Arrays
- 1.6.2. Matrices. Poliedros
- 1.6.3. Búsqueda y ordenación
- 1.6.4. Cadenas. Funciones de E/S para cadenas
- 1.6.5. Estructuras. Uniones
- 1.6.6. Nuevos tipos de datos

1.7. Estructuras de datos dinámicas: punteros
- 1.7.1. Concepto. Definición de puntero
- 1.7.2. Operadores y operaciones con punteros
- 1.7.3. Arrays de punteros
- 1.7.4. Punteros y Arrays
- 1.7.5. Punteros a cadenas
- 1.7.6. Punteros a estructuras
- 1.7.7. Indirección múltiple
- 1.7.8. Punteros a funciones
- 1.7.9. Paso de funciones, estructuras y arrays como parámetros de funciones

1.8. Ficheros
- 1.8.1. Conceptos básicos
- 1.8.2. Operaciones con ficheros
- 1.8.3. Tipos de ficheros
- 1.8.4. Organización de los ficheros
- 1.8.5. Introducción a los ficheros C++
- 1.8.6. Manejo de ficheros

1.9. Recursividad
- 1.9.1. Definición de recursividad
- 1.9.2. Tipos de recursión
- 1.9.3. Ventajas e inconvenientes
- 1.9.4. Consideraciones
- 1.9.5. Conversión recursivo-iterativa
- 1.9.6. La pila de recursión

1.10. Prueba y documentación
- 1.10.1. Pruebas de programas
- 1.10.2. Prueba de la caja blanca
- 1.10.3. Prueba de la caja negra
- 1.10.4. Herramientas para realizar las pruebas
- 1.10.5. Documentación de programas

Módulo 2. Estructura de datos

2.1. Introducción a la programación en C++
- 2.1.1. Clases, constructores, métodos y atributos
- 2.1.2. Variables
- 2.1.3. Expresiones condicionales y bucles
- 2.1.4. Objetos

2.2. Tipos Abstractos de Datos (TAD)
- 2.2.1. Tipos de datos
- 2.2.2. Estructuras básicas y TAD
- 2.2.3. Vectores y Arrays

2.3. Estructuras de datos lineales
- 2.3.1. TAD lista. Definición
- 2.3.2. Listas enlazadas y doblemente enlazadas
- 2.3.3. Listas ordenadas
- 2.3.4. Listas en C++
- 2.3.5. TAD Pila
- 2.3.6. TAD Cola
- 2.3.7. Pila y Cola en C++

2.4. Estructuras de datos jerárquicas
- 2.4.1. TAD Árbol
- 2.4.2. Recorridos
- 2.4.3. Árboles n-arios
- 2.4.4. Árboles binarios
- 2.4.5. Árboles binarios de búsqueda

2.5. Estructuras de datos jerárquicas: árboles complejos
- 2.5.1. Árboles perfectamente equilibrados o de altura mínima
- 2.5.2. Árboles multicamino
- 2.5.3. Referencias bibliográficas

2.6. Montículos y cola de prioridad
- 2.6.1. TAD Montículos
- 2.6.2. TAD Cola de prioridad

2.7. Tablas hash
- 2.7.1. TAD Tabla *Hash*
- 2.7.2. Funciones *Hash*
- 2.7.3. Función *Hash* en tablas hash
- 2.7.4. Redispersión
- 2.7.5. Tablas hash abiertas

2.8. Grafos
- 2.8.1. TAD Grafo
- 2.8.2. Tipos de grafo
- 2.8.3. Representación gráfica y operaciones básicas
- 2.8.4. Diseño de grafos

2.9. Algoritmos y conceptos avanzados sobre grafos
- 2.9.1. Problemas sobre grafos
- 2.9.2. Algoritmos sobre caminos
- 2.9.3. Algoritmos de búsqueda o recorridos
- 2.9.4. Otros algoritmos

2.10. Otras estructuras de datos
- 2.10.1. Conjuntos
- 2.10.2. Arrays paralelos
- 2.10.3. Tablas de símbolos
- 2.10.4. Tries

Módulo 3. Algoritmia y complejidad

3.1. Introducción a las estrategias de diseño de algoritmos
- 3.1.1. Recursividad
- 3.1.2. Divide y conquista
- 3.1.3. Otras estrategias

3.2. Eficiencia y análisis de los algoritmos
- 3.2.1. Medidas de eficiencia
- 3.2.2. Medir el tamaño de la entrada
- 3.2.3. Medir el tiempo de ejecución
- 3.2.4. Caso peor, mejor y medio
- 3.2.5. Notación asintónica
- 3.2.6. Criterios de Análisis matemático de algoritmos no recursivos
- 3.2.7. Análisis matemático de algoritmos recursivos
- 3.2.8. Análisis empírico de algoritmos

3.3. Algoritmos de ordenación
- 3.3.1. Concepto de ordenación
- 3.3.2. Ordenación de la burbuja
- 3.3.3. Ordenación por selección
- 3.3.4. Ordenación por inserción
- 3.3.5. Ordenación por mezcla (*Merge_Sort)*
- 3.3.6. Ordenación rápida (*Quick_Sort*)

3.4 Algoritmos con árboles
- 3.4.1. Concepto de árbol
- 3.4.2. Árboles binarios
- 3.4.3. Recorridos de árbol
- 3.4.4. Representar expresiones
- 3.4.5. Árboles binarios ordenados
- 3.4.6. Árboles binarios balanceados

3.5. Algoritmos con *Heaps*
- 3.5.1. Los *Heaps*
- 3.5.2. El algoritmo *Heapsort*
- 3.5.3. Las colas de prioridad



3.6. Algoritmos con grafos
- 3.6.1. Representación
- 3.6.2. Recorrido en anchura
- 3.6.3. Recorrido en profundidad
- 3.6.4. Ordenación topológica

3.7. Algoritmos *Greedy*
- 3.7.1. La estrategia *Greedy*
- 3.7.2. Elementos de la estrategia *Greedy*
- 3.7.3. Cambio de monedas
- 3.7.4. Problema del viajante
- 3.7.5. Problema de la mochila

3.8. Búsqueda de caminos mínimos
- 3.8.1. El problema del camino mínimo
- 3.8.2. Arcos negativos y ciclos
- 3.8.3. Algoritmo de Dijkstra

3.9. Algoritmos *Greedy* sobre grafos
- 3.9.1. El árbol de recubrimiento mínimo
- 3.9.2. El algoritmo de Prim
- 3.9.3. El algoritmo de Kruskal
- 3.9.4. Análisis de complejidad

3.10. *Backtracking*
- 3.10.1. El *Backtracking*
- 3.10.2. Técnicas alternativas

Módulo 4. Diseño avanzado de algoritmos

4.1. Análisis de algoritmos recursivos y tipo divide y conquista
- 4.1.1. Planteamiento y resolución de ecuaciones de recurrencia homogéneas y no homogéneas.
- 4.1.2. Descripción general de la estrategia divide y conquista

4. 2. Análisis amortizado
- 4.2.1. El análisis agregado
- 4.2.2. El método de contabilidad
- 4.2.3. El método del potencial

4.3. Programación dinámica y algoritmos para problemas NP
- 4.3.1. Características de la programación dinámica
- 4.3.2. Vuelta atrás: backtracking
- 4.3.3. Ramificación y poda

4.4. Optimización combinatoria
- 4.4.1. Representación de problemas
- 4.4.2. Optimización en 1D

4.5. Algoritmos de aleatorización
- 4.5.1. Ejemplos de algoritmos de aleatorización
- 4.5.2. El teorema Buffon
- 4.5.3. Algoritmo de Monte Carlo
- 4.5.4. Algoritmo Las Vegas

4.6. Búsqueda local y con candidatos
- 4.6.1. Garcient *Ascent*
- 4.6.2. Hill *Climbing*
- 4.6.3. Simulated *Annealing*
- 4.6.4. Tabu *Search*
- 4.6.5. Búsqueda con candidatos

4.7. Verificación formal de programas
- 4.7.1. Especificación de abstracciones funcionales
- 4.7.2. El lenguaje de la lógica de primer orden
- 4.7.3. El sistema formal de Hoare

4.8. Verificación de programas iterativos
- 4.8.1. Reglas del sistema formal de Hoare
- 4.8.2. Concepto de invariante de iteraciones

4.9. Métodos numéricos
- 4.9.1. El método de la bisección
- 4.9.2. El método de Newton Raphson
- 4.9.3. El método de la secante

4.10. Algoritmos paralelos
- 4.10.1. Operaciones binarias paralelas
- 4.10.2. Operaciones paralelas con grafos
- 4.10.3. Paralelismo en divide y vencerás
- 4.10.4. Paralelismo en programación dinámica

Módulo 5. Lógica computacional

5.1. Justificación de la lógica
- 5.1.1. Objeto del estudio de la lógica
- 5.1.2. ¿Para qué sirve la lógica?
- 5.1.3. Componentes y tipos de razonamiento
- 5.1.4. Componentes de un cálculo lógico
- 5.1.5. Semántica
- 5.1.6. Justificación de la existencia de una lógica
- 5.1.7. ¿Cómo comprobar que una lógica es adecuada?

5.2. Cálculo de deducción natural de enunciados
- 5.2.1. Lenguaje formal
- 5.2.2. Mecanismo deductivo

5.3. Estrategias de formalización y deducción para la lógica proposicional
- 5.3.1. Estrategias de formalización
- 5.3.2. El razonamiento natural
- 5.3.3. Leyes y reglas
- 5.3.4. Deducción axiomática y deducción natural
- 5.3.5. El cálculo de la deducción natural
- 5.3.6. Reglas primitivas del cálculo proposicional

5.4. Semántica de la lógica proposicional
- 5.4.1. Tablas de verdad
- 5.4.2. Equivalencia
- 5.4.3. Tautologías y contradicciones
- 5.4.4. Validación de sentencias proposicionales
- 5.4.5. Validación mediante tablas de verdad
- 5.4.6. Validación mediante árboles semánticos
- 5.4.7. Validación mediante refutación

5.5. Aplicaciones de la lógica proposicional: circuitos lógicos
- 5.5.1. Las puertas básicas
- 5.5.2. Circuitos
- 5.5.3. Modelos matemáticos de los circuitos
- 5.5.4. Minimización
- 5.5.5. La segunda forma canónica y la forma mínima en producto de sumas
- 5.5.6. Otras puertas

5.6. Cálculo de deducción natural de predicados
- 5.6.1. Lenguaje formal
- 5.6.2. Mecanismo deductivo

5.7. Estrategias de formalización para la lógica de predicados
- 5.7.1. Introducción a la formalización en lógica de predicados
- 5.7.2. Estrategias de formalización con cuantificadores

5.8. Estrategias de deducción para la lógica de predicados
- 5.8.1. Razón de una omisión
- 5.8.2. Presentación de las nuevas reglas
- 5.8.3. La lógica de predicados como cálculo de deducción natural

5.9. Aplicaciones de la lógica de predicados: introducción a la programación lógica
- 5.9.1. Presentación informal
- 5.9.2. Elementos del Prolog
- 5.9.3. La reevaluación y el corte

5.10. Teoría de conjuntos, lógica de predicados y su semántica
- 5.10.1. Teoría intuitiva de conjuntos
- 5.10.2. Introducción a la semántica de predicados

Módulo 6. Inteligencia artificial e Ingeniería del Conocimiento

6.1. Introducción a la Inteligencia Artificial y a la Ingeniería del Conocimiento
- 6.1.1. Breve historia de la Inteligencia Artificial
- 6.1.2. La Inteligencia Artificial hoy en día
- 6.1.3. Ingeniería del Conocimiento

6.2. Búsqueda
- 6.2.1. Conceptos comunes de búsqueda
- 6.2.2. Búsqueda no informada
- 6.2.3. Búsqueda informada

6.3. Satisfacibilidad booleana, satisfacibilidad de restricciones y planificación automática
- 6.3.1. Satisfacibilidad booleana
- 6.3.2. Problemas de satisfacción de restricciones
- 6.3.3. Planificación automática y PDDL
- 6.3.4. Planificación como búsqueda heurística
- 6.3.5. Planificación con SAT

6.4. La Inteligencia Artificial en juegos
- 6.4.1. Teoría de Juegos
- 6.4.2. Minimax y poda Alfa-Beta
- 6.4.3. Simulación: Monte Carlo

6.5. Aprendizaje supervisado y no supervisado
- 6.5.1. Introducción al aprendizaje automático
- 6.5.2. Clasificación
- 6.5.3. Regresión
- 6.5.4. Validación de resultados
- 6.5.5. Agrupación (*Clustering*)

6.6. Redes de neuronas
- 6.6.1. Fundamentos biológicos
- 6.6.2. Modelo computacional
- 6.6.3. Redes de neuronas supervisadas y no supervisadas
- 6.6.4. Perceptrón simple
- 6.6.5. Perceptrón multicapa

6.7. Algoritmos genéticos
- 6.7.1. Historia
- 6.7.2. Base biológica
- 6.7.3. Codificación de problemas
- 6.7.4. Generación de la población inicial
- 6.7.5. Algoritmo principal y operadores genéticos
- 6.7.6. Evaluación de individuos: *Fitness*

6.8. Tesauros, vocabularios, taxonomías
- 6.8.1. Vocabularios
- 6.8.2. Taxonomías
- 6.8.3. Tesauros
- 6.8.4. Ontologías

6.9. Representación del conocimiento: web semántica
- 6.9.1. Web semántica
- 6.9.2. Especificaciones: RDF, RDFS y OWL
- 6.9.3. Inferencia/razonamiento
- 6.9.4. Linked Data

6.10. Sistemas expertos y DSS
- 6.10.1. Sistemas expertos
- 6.10.2. Sistemas de soporte a la decisión

Módulo 7. Sistemas inteligentes

7.1. Teoría de agentes
- 7.1.1. Historia del concepto
- 7.1.2. Definición de agente
- 7.1.3. Agentes en Inteligencia Artificial
- 7.1.4. Agentes en ingeniería de Software

7.2. Arquitecturas de agentes
- 7.2.1. El proceso de razonamiento de un agente
- 7.2.2. Agentes reactivos
- 7.2.3. Agentes deductivos
- 7.2.4. Agentes híbridos
- 7.2.5. Comparativa

7.3. Información y conocimiento
- 7.3.1. Distinción entre datos, información y conocimiento
- 7.3.2. Evaluación de la calidad de los datos
- 7.3.3. Métodos de captura de datos
- 7.3.4. Métodos de adquisición de información
- 7.3.5. Métodos de adquisición de conocimiento

7.4. Representación del conocimiento
- 7.4.1. La importancia de la representación del conocimiento
- 7.4.2. Definición de representación del conocimiento a través de sus roles
- 7.4.3. Características de una representación del conocimiento

7.5. Ontologías
- 7.5.1. Introducción a los metadatos
- 7.5.2. Concepto filosófico de ontología
- 7.5.3. Concepto informático de ontología
- 7.5.4. Ontologías de dominio y ontologías de nivel superior
- 7.5.5. ¿Cómo construir una ontología?

7.6. Lenguajes para ontologías y software para la creación de ontologías
- 7.6.1. Tripletas RDF, Turtle y N3
- 7.6.2. RDF Schema
- 7.6.3. OWL
- 7.6.4. SPARQL
- 7.6.5. Introducción a las diferentes herramientas para la creación de ontologías
- 7.6.6. Instalación y uso de Protégé

7.7. La web semántica
- 7.7.1. El estado actual y futuro de la web semántica
- 7.7.2. Aplicaciones de la web semántica

7.8. Otros modelos de representación del conocimiento
- 7.8.1. Vocabularios
- 7.8.2. Visión global
- 7.8.3. Taxonomías
- 7.8.4. Tesauros
- 7.8.5. Folksonomías
- 7.8.6. Comparativa
- 7.8.7. Mapas mentales

7.9. Evaluación e integración de representaciones del conocimiento
- 7.9.1. Lógica de orden cero
- 7.9.2. Lógica de primer orden
- 7.9.3. Lógica descriptiva
- 7.9.4. Relación entre diferentes tipos de lógica
- 7.9.5. Prolog: programación basada en lógica de primer orden

7.10. Razonadores semánticos, sistemas basados en conocimiento y Sistemas Expertos
- 7.10.1. Concepto de razonador
- 7.10.2. Aplicaciones de un razonador
- 7.10.3. Sistemas basados en el conocimiento
- 7.10.4. MYCIN, historia de los Sistemas Expertos
- 7.10.5. Elementos y Arquitectura de Sistemas Expertos
- 7.10.6. Creación de Sistemas Expertos

Módulo 8. Aprendizaje automático y minería de datos

8.1. Introducción a los procesos de descubrimiento del conocimiento y conceptos básicos de aprendizaje automático
- 8.1.1. Conceptos clave de los procesos de descubrimiento del conocimiento
- 8.1.2. Perspectiva histórica de los procesos de descubrimiento del conocimiento
- 8.1.3. Etapas de los procesos de descubrimiento del conocimiento
- 8.1.4. Técnicas utilizadas en los procesos de descubrimiento del conocimiento
- 8.1.5. Características de los buenos modelos de aprendizaje automático
- 8.1.6. Tipos de información de aprendizaje automático
- 8.1.7. Conceptos básicos de aprendizaje
- 8.1.8. Conceptos básicos de aprendizaje no supervisado

8.2. Exploración y preprocesamiento de datos
- 8.2.1. Tratamiento de datos
- 8.2.2. Tratamiento de datos en el flujo de análisis de datos
- 8.2.3. Tipos de datos
- 8.2.4. Transformaciones de datos
- 8.2.5. Visualización y exploración de variables continuas
- 8.2.6. Visualización y exploración de variables categóricas
- 8.2.7. Medidas de correlación
- 8.2.8. Representaciones gráficas más habituales
- 8.2.9. Introducción al análisis multivariante y a la reducción de dimensiones

8.3. Árboles de decisión
- 8.3.1. Algoritmo ID3
- 8.3.2. Algoritmo C4.5
- 8.3.3. Sobreentrenamiento y poda
- 8.3.4. Análisis de resultados

8.4. Evaluación de clasificadores
- 8.4.1. Matrices de confusión
- 8.4.2. Matrices de evaluación numérica
- 8.4.3. Estadístico de Kappa
- 8.4.5. La curva ROC

8.5. Reglas de clasificación
- 8.5.1. Medidas de evaluación de reglas
- 8.5.2. Introducción a la representación gráfica
- 8.5.3. Algoritmo de recubrimiento secuencial

8.6. Redes neuronales
- 8.6.1. Conceptos básicos
- 8.6.2. Redes de neuronas simples
- 8.6.3. Algoritmo de *Backpropagation*
- 8.6.4. Introducción a las redes neuronales recurrentes

8.7. Métodos bayesianos
- 8.7.1. Conceptos básicos de probabilidad
- 8.7.2. Teorema de Bayes
- 8.7.3. Naive Bayes
- 8.7.4. Introducción a las redes bayesianas

8.8. Modelos de regresión y de respuesta continua
- 8.8.1. Regresión lineal simple
- 8.8.2. Regresión lineal múltiple
- 8.8.3. Regresión logística
- 8.8.4. Árboles de regresión
- 8.8.5. Introducción a las máquinas de soporte vectorial (SVM)
- 8.8.6. Medidas de bondad de ajuste

8.9. *Clustering*
- 8.9.1. Conceptos básicos
- 8.9.2. Clustering jerárquico
- 8.9.3. Métodos probabilistas
- 8.9.4. Algoritmo EM
- 8.9.5. Método B-Cubed
- 8.9.6. Métodos implícitos

8.10. Minería de textos y procesamiento de lenguaje natural (NLP)
- 8.10.1. Conceptos básicos
- 8.10.2. Creación del corpus
- 8.10.3. Análisis descriptivo
- 8.10.4. Introducción al análisis de sentimientos

Módulo 9. Sistemas multiagente y percepción computacional

9.1. Agentes y sistemas multiagente
- 9.1.1. Concepto de agente
- 9.1.2. Arquitecturas
- 9.1.3. Comunicación y coordinación
- 9.1.4. Lenguajes de programación y herramientas
- 9.1.5. Aplicaciones de los agentes
- 9.1.6. La FIPA

9.2. El estándar para agentes: FIPA
- 9.2.1. La comunicación entre los agentes
- 9.2.2. La gestión de los agentes
- 9.2.3. La arquitectura abstracta
- 9.2.4. Otras especificaciones

9.3. La plataforma JADE
- 9.3.1. Los agentes software según JADE
- 9.3.2. Arquitectura
- 9.3.3. Instalación y ejecución
- 9.3.4. Paquetes JADE

9.4. Programación básica con JADE
- 9.4.1. La consola de gestión
- 9.4.2. Creación básica de agentes

9.5. Programación avanzada con JADE
- 9.5.1. Creación avanzada de agentes
- 9.5.2. Comunicación entre agentes
- 9.5.3. Descubrimiento de agentes

9.6. Visión artificial
- 9.6.1. Procesamiento y análisis digital de imágenes
- 9.6.2. Análisis de imágenes y visión artificial
- 9.6.3. Procesamiento de imágenes y visión humana
- 9.6.4. Sistema de capturas de imágenes
- 9.6.5. Formación de la imagen y percepción

9.7. Análisis de imágenes digitales
- 9.7.1. Etapas del proceso de análisis de imágenes
- 9.7.2. Preprocesado
- 9.7.3. Operaciones básicas
- 9.7.4. Filtrado espacial

9.8. Transformación de imágenes digitales y segmentación de imágenes
- 9.8.1. Transformadas de Fourier
- 9.8.2. Filtrado en frecuencias
- 9.8.3. Conceptos básicos
- 9.8.4. Umbralización
- 9.8.5. Detección de contornos

9.9. Reconocimiento de formas
- 9.9.1. Extracción de características
- 9.9.2. Algoritmos de clasificación

9.10. Procesamiento de lenguaje natural
- 9.10.1. Reconocimiento automático del habla
- 9.10.2. Lingüística computacional

Módulo 10. Computación bioinspirada

10.1. Introducción a la computación bioinspirada
- 10.1.1. Introducción a la computación bioinspirada

10.2. Algoritmos de adaptación social
- 10.2.1. Computación bioinspirada basada en colonia de hormigas
- 10.2.2. Variantes de los algoritmos de colonias de hormigas
- 10.2.3. Computación basada en nubes de partículas

10.3. Algoritmos genéticos
- 10.3.1. Estructura general
- 10.3.2. Implementaciones de los principales operadores

10.4. Estrategias de exploración-explotación del espacio para algoritmos genéticos
- 10.4.1. Algoritmo CHC
- 10.4.2. Problemas multimodales

10.5. Modelos de computación evolutiva (I)
- 10.5.1. Estrategias evolutivas
- 10.5.2. Programación evolutiva
- 10.5.3. Algoritmos basados en evolución diferencial

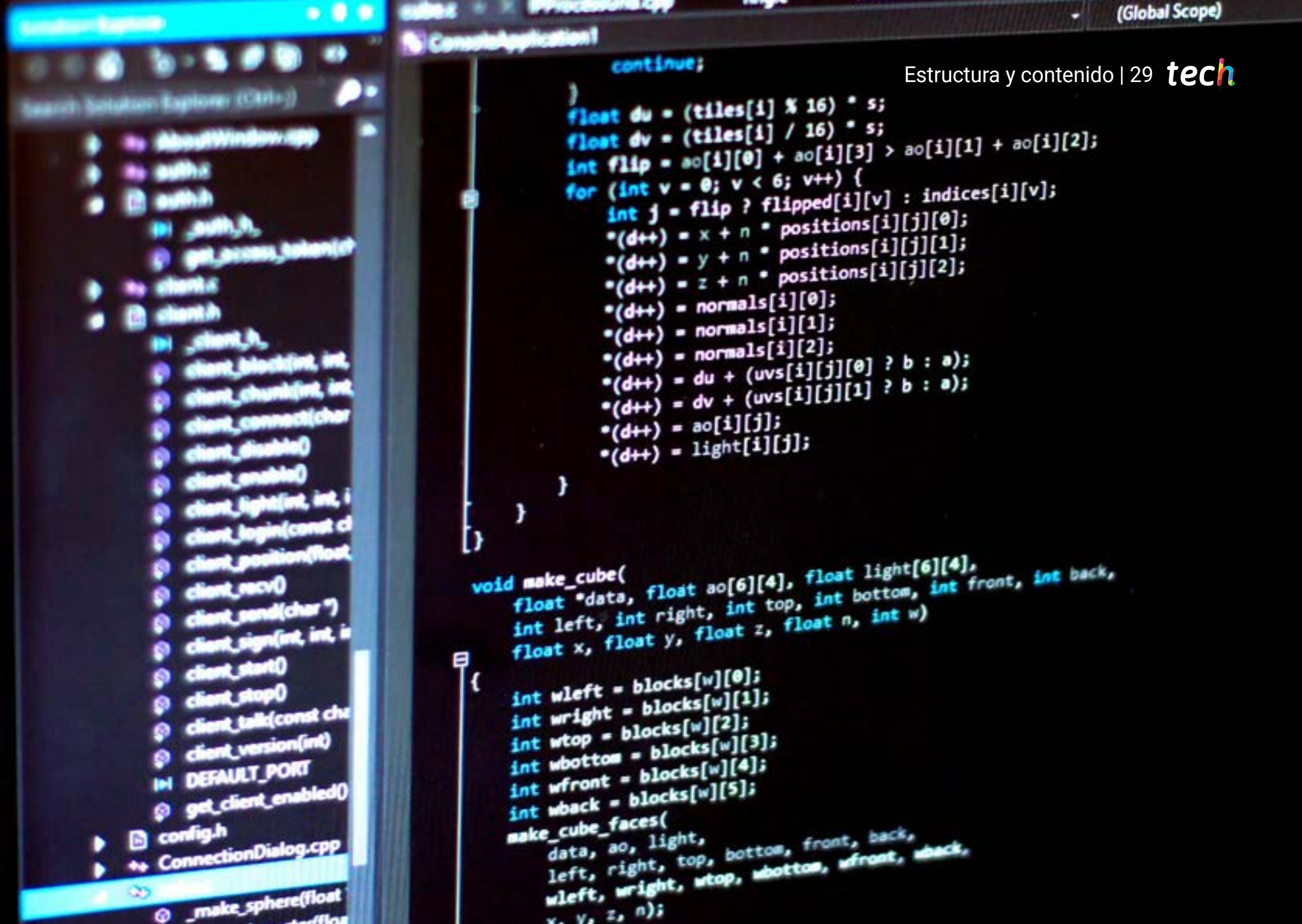
10.6. Modelos de computación evolutiva (II)
- 10.6.1. Modelos de evolución basados en estimación de distribuciones (EDA)
- 10.6.2. Programación genética

10.7. Programación evolutiva aplicada a problemas de aprendizaje
- 10.7.1. Aprendizaje basado en reglas
- 10.7.2. Métodos evolutivos en problemas de selección de instancias

10.8. Problemas multiobjetivo
- 10.8.1. Concepto de dominancia
- 10.8.2. Aplicación de algoritmos evolutivos a problemas multiobjetivo

10.9. Redes neuronales (I)
- 10.9.1. Introducción a las redes neuronales
- 10.9.2. Ejemplo práctico con redes neuronales

10.10. Redes neuronales (II)
- 10.10.1. Casos de uso de las redes neuronales en la investigación médica
- 10.10.2. Casos de uso de las redes neuronales en la economía
- 10.10.3. Casos de uso de las redes neuronales en la visión artificial


(Global Scope)
ConsoleApplication1
continue;
}
float du = (tiles[i] % 16) * s;
float dv = (tiles[i] / 16) * s;
int flip = ao[i][0] + ao[i][3] > ao[i][1] + ao[i][2];
for (int v = 0; v < 6; v++) {
int j = flip ? flipped[i][v] : indices[i][v];
*(d++) = x + n * positions[i][j][0];
*(d++) = y + n * positions[i][j][1];
*(d++) = z + n * positions[i][j][2];
*(d++) = normals[i][0];
*(d++) = normals[i][1];
*(d++) = normals[i][2];
*(d++) = du + (uvs[i][j][0] ? b : a);
*(d++) = dv + (uvs[i][j][1] ? b : a);
*(d++) = ao[i][j];
*(d++) = light[i][j];
}
}
}
void make_cube(
float *data, float ao[6][4], float light[6][4],
int left, int right, int top, int bottom, int front, int back,
float x, float y, float z, float n, int w)
{
int wleft = blocks[w][0];
int wright = blocks[w][1];
int wtop = blocks[w][2];
int wbottom = blocks[w][3];
int wfront = blocks[w][4];
int wback = blocks[w][5];
make_cube_faces(
data, ao, light,
left, right, top, bottom, front, back,
wleft, wright, wtop, wbottom, wfront, wback,
x, y, z, n);
AboutWindow.cpp
auth.c
auth.h
_auth_h_
client.c
client.h
_client_h_
client_disable()
client_enable()
client_recv()
client_send(char *)
client_start()
client_stop()
client_version(int)
DEFAULT_PORT
get_client_enabled()
config.h
ConnectionDialog.cpp
_make_sphere(float

05

Metodología

Este programa de capacitación ofrece una forma diferente de aprender. Nuestra metodología se desarrolla a través de un modo de aprendizaje de forma cíclica: ***el Relearning***.

Este sistema de enseñanza es utilizado, por ejemplo, en las facultades de medicina más prestigiosas del mundo y se ha considerado uno de los más eficaces por publicaciones de gran relevancia como el ***New England Journal of Medicine.***





Descubre el Relearning, un sistema que abandona el aprendizaje lineal convencional para llevarte a través de sistemas cíclicos de enseñanza: una forma de aprender que ha demostrado su enorme eficacia, especialmente en las materias que requieren memorización"

Estudio de Caso para contextualizar todo el contenido

Nuestro programa ofrece un método revolucionario de desarrollo de habilidades y conocimientos. Nuestro objetivo es afianzar competencias en un contexto cambiante, competitivo y de alta exigencia.

Con TECH podrás experimentar una forma de aprender que está moviendo los cimientos de las universidades tradicionales de todo el mundo"



Accederás a un sistema de aprendizaje basado en la reiteración, con una enseñanza natural y progresiva a lo largo de todo el temario.



El alumno aprenderá, mediante actividades colaborativas y casos reales, la resolución de situaciones complejas en entornos empresariales reales.

Un método de aprendizaje innovador y diferente

El presente programa de TECH es una enseñanza intensiva, creada desde 0, que propone los retos y decisiones más exigentes en este campo, ya sea en el ámbito nacional o internacional. Gracias a esta metodología se impulsa el crecimiento personal y profesional, dando un paso decisivo para conseguir el éxito. El método del caso, técnica que sienta las bases de este contenido, garantiza que se sigue la realidad económica, social y profesional más vigente.

> *Nuestro programa te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera"*

El método del caso ha sido el sistema de aprendizaje más utilizado por las mejores escuelas de Informática del mundo desde que éstas existen. Desarrollado en 1912 para que los estudiantes de Derecho no solo aprendiesen las leyes a base de contenidos teóricos, el método del caso consistió en presentarles situaciones complejas reales para que tomasen decisiones y emitiesen juicios de valor fundamentados sobre cómo resolverlas. En 1924 se estableció como método estándar de enseñanza en Harvard.

Ante una determinada situación, ¿qué debería hacer un profesional? Esta es la pregunta a la que te enfrentamos en el método del caso, un método de aprendizaje orientado a la acción. A lo largo del curso, los estudiantes se enfrentarán a múltiples casos reales. Deberán integrar todos sus conocimientos, investigar, argumentar y defender sus ideas y decisiones.

Relearning Methodology

TECH aúna de forma eficaz la metodología del Estudio de Caso con un sistema de aprendizaje 100% online basado en la reiteración, que combina elementos didácticos diferentes en cada lección.

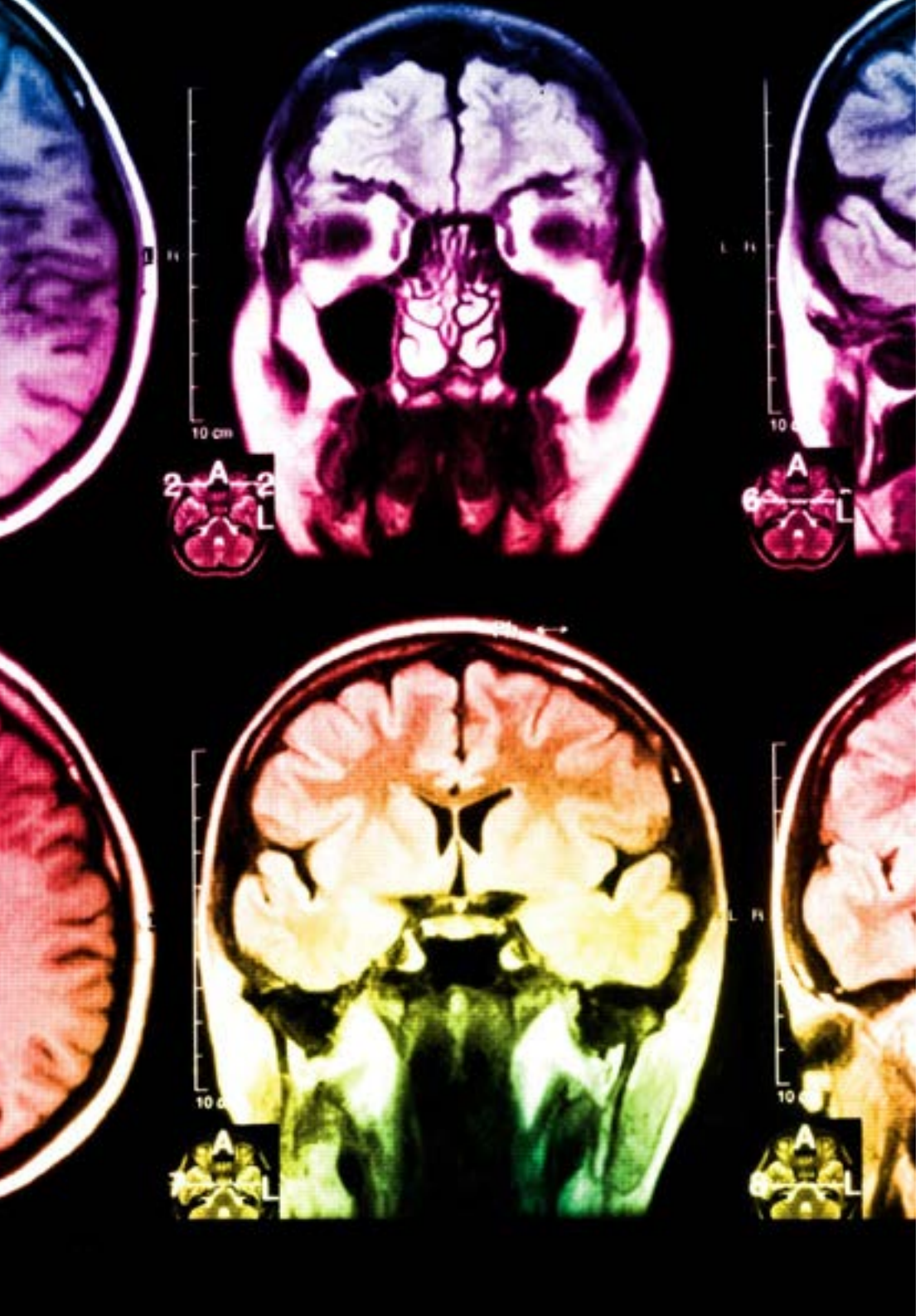
Potenciamos el Estudio de Caso con el mejor método de enseñanza 100% online: el Relearning.

En 2019 obtuvimos los mejores resultados de aprendizaje de todas las universidades online en español en el mundo.

En TECH aprenderás con una metodología vanguardista concebida para capacitar a los directivos del futuro. Este método, a la vanguardia pedagógica mundial, se denomina Relearning.

Nuestra universidad es la única en habla hispana licenciada para emplear este exitoso método. En 2019, conseguimos mejorar los niveles de satisfacción global de nuestros alumnos (calidad docente, calidad de los materiales, estructura del curso, objetivos…) con respecto a los indicadores de la mejor universidad online en español.





En nuestro programa, el aprendizaje no es un proceso lineal, sino que sucede en espiral (aprender, desaprender, olvidar y reaprender). Por eso, se combinan cada uno de estos elementos de forma concéntrica. Con esta metodología se han capacitado más de 650.000 graduados universitarios con un éxito sin precedentes en ámbitos tan distintos como la bioquímica, la genética, la cirugía, el derecho internacional, las habilidades directivas, las ciencias del deporte, la filosofía, el derecho, la ingeniería, el periodismo, la historia o los mercados e instrumentos financieros. Todo ello en un entorno de alta exigencia, con un alumnado universitario de un perfil socioeconómico alto y una media de edad de 43,5 años.

El Relearning te permitirá aprender con menos esfuerzo y más rendimiento, implicándote más en tu capacitación, desarrollando el espíritu crítico, la defensa de argumentos y el contraste de opiniones: una ecuación directa al éxito.

A partir de la última evidencia científica en el ámbito de la neurociencia, no solo sabemos organizar la información, las ideas, las imágenes y los recuerdos, sino que sabemos que el lugar y el contexto donde hemos aprendido algo es fundamental para que seamos capaces de recordarlo y almacenarlo en el hipocampo, para retenerlo en nuestra memoria a largo plazo.

De esta manera, y en lo que se denomina Neurocognitive context-dependent e-learning, los diferentes elementos de nuestro programa están conectados con el contexto donde el participante desarrolla su práctica profesional.

Este programa ofrece los mejores materiales educativos, preparados a conciencia para los profesionales:

Material de estudio

Todos los contenidos didácticos son creados por los especialistas que van a impartir el curso, específicamente para él, de manera que el desarrollo didáctico sea realmente específico y concreto.

Estos contenidos son aplicados después al formato audiovisual, para crear el método de trabajo online de TECH. Todo ello, con las técnicas más novedosas que ofrecen piezas de gran calidad en todos y cada uno los materiales que se ponen a disposición del alumno.

Clases magistrales

Existe evidencia científica sobre la utilidad de la observación de terceros expertos.

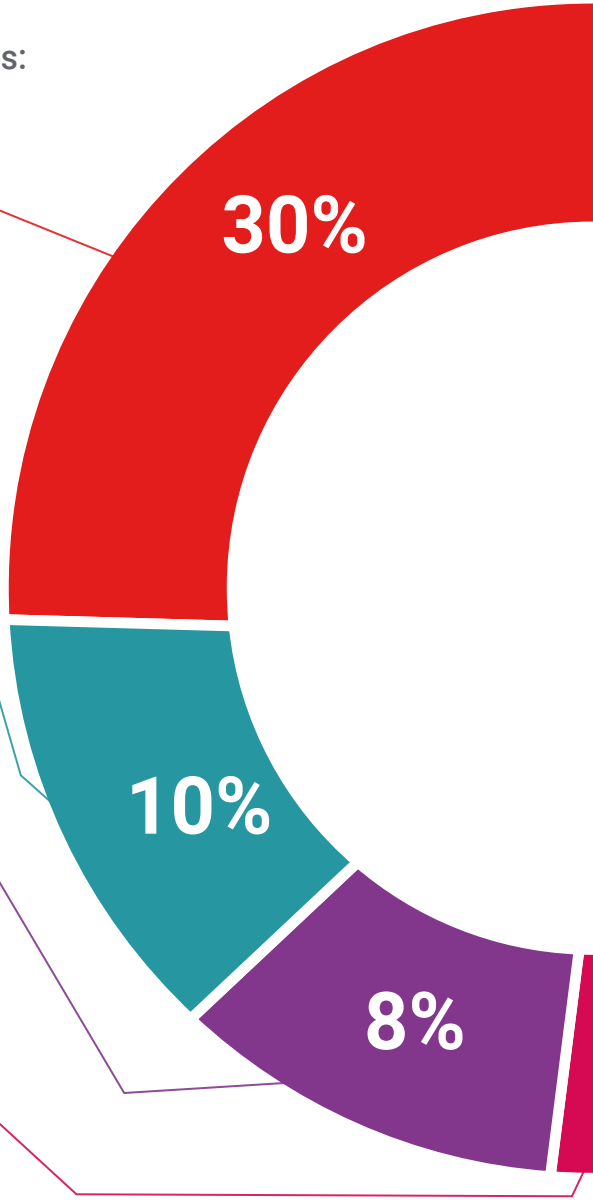
El denominado Learning from an Expert afianza el conocimiento y el recuerdo, y genera seguridad en las futuras decisiones difíciles.

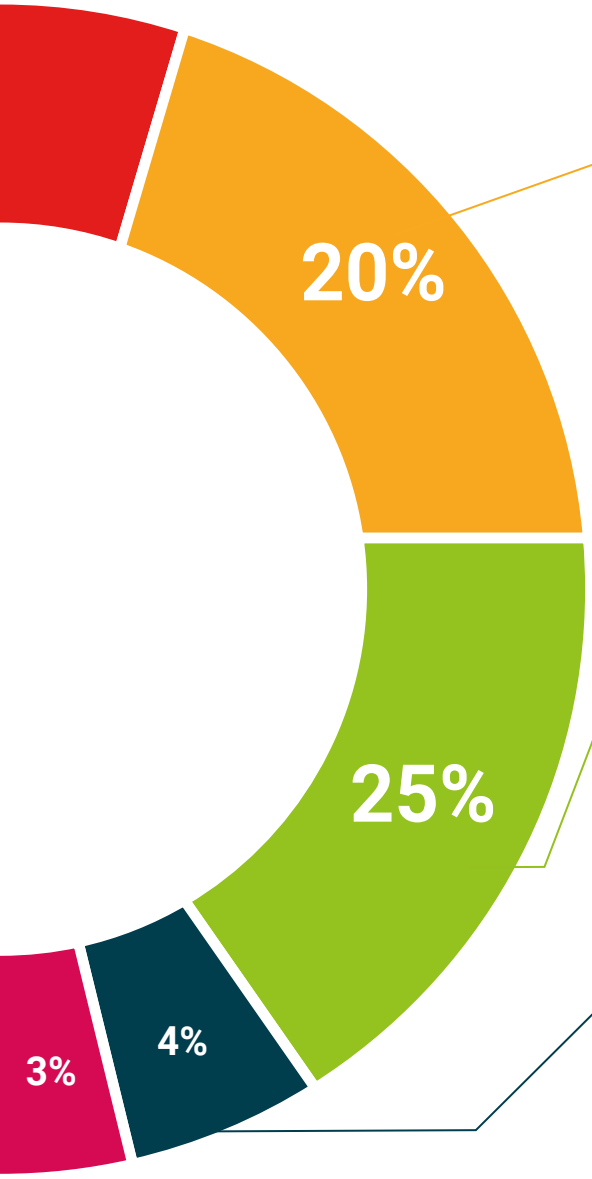
Prácticas de habilidades y competencias

Realizarán actividades de desarrollo de competencias y habilidades específicas en cada área temática. Prácticas y dinámicas para adquirir y desarrollar las destrezas y habilidades que un especialista precisa desarrollar en el marco de la globalización que vivimos.

Lecturas complementarias

Artículos recientes, documentos de consenso y guías internacionales, entre otros. En la biblioteca virtual de TECH el estudiante tendrá acceso a todo lo que necesita para completar su capacitación.





Case studies

Completarán una selección de los mejores casos de estudio elegidos expresamente para esta titulación. Casos presentados, analizados y tutorizados por los mejores especialistas del panorama internacional.

Resúmenes interactivos

El equipo de TECH presenta los contenidos de manera atractiva y dinámica en píldoras multimedia que incluyen audios, vídeos, imágenes, esquemas y mapas conceptuales con el fin de afianzar el conocimiento.

Este exclusivo sistema educativo para la presentación de contenidos multimedia fue premiado por Microsoft como "Caso de éxito en Europa".

Testing & Retesting

Se evalúan y reevalúan periódicamente los conocimientos del alumno a lo largo del programa, mediante actividades y ejercicios evaluativos y autoevaluativos para que, de esta manera, el estudiante compruebe cómo va consiguiendo sus metas.

06

Titulación

El Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Máster Propio expedido por TECH Universidad Tecnológica.





Supera con éxito este programa y recibe tu titulación universitaria sin desplazamientos ni farragosos trámites”

Este **Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento** contiene el programa más completo y actualizado del mercado.

Tras la superación de la evaluación, el alumno recibirá por correo postal* con acuse de recibo su correspondiente título de **Máster Propio** emitido por **TECH Universidad Tecnológica.**

El título expedido por **TECH Universidad Tecnológica** expresará la calificación que haya obtenido en el Máster Título Propio, y reunirá los requisitos comúnmente exigidos por las bolsas de trabajo, oposiciones y comités evaluadores de carreras profesionales.

Título: **Máster Título Propio Inteligencia Artificial e Ingeniería del Conocimiento**
N.º Horas Oficiales: **1.500 h.**


tech universidad tecnológica

Otorga la presente

CONSTANCIA

a

C. _____, con documento de identificación nº_____

Por haber superado con éxito y acreditado el programa de

MÁSTER TÍTULO PROPIO

en

Inteligencia Artificial e Ingeniería del Conocimiento

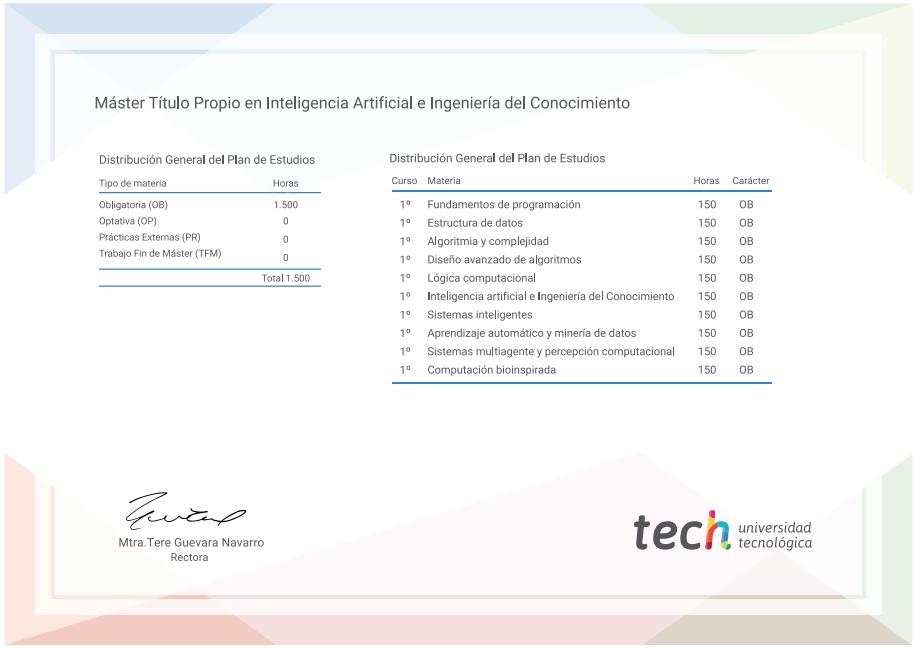
Se trata de un título propio de esta Universidad con una duración de 1.500 horas, con fecha de inicio dd/mm/aaaa y fecha de finalización dd/mm/aaaa.

TECH es una Institución Particular de Educación Superior reconocida por la Secretaría de Educación Pública a partir del 28 de junio de 2018.

A 17 de junio de 2020

Mtra.Tere Guevara Navarro
Rectora

Este título propio se deberá acompañar siempre del título universitario habilitante expedido por la autoridad competente para ejercer profesionalmente en cada país. código único TECH: AFWOR23S techtitute.com/titulos


Máster Título Propio en Inteligencia Artificial e Ingeniería del Conocimiento

Distribución General del Plan de Estudios

Tipo de materia	Horas
Obligatoria (OB)	1.500
Optativa (OP)	0
Prácticas Externas (PR)	0
Trabajo Fin de Máster (TFM)	0
	Total 1.500

Distribución General del Plan de Estudios

Curso	Materia	Horas	Carácter
1º	Fundamentos de programación	150	OB
1º	Estructura de datos	150	OB
1º	Algoritmia y complejidad	150	OB
1º	Diseño avanzado de algoritmos	150	OB
1º	Lógica computacional	150	OB
1º	Inteligencia artificial e Ingeniería del Conocimiento	150	OB
1º	Sistemas inteligentes	150	OB
1º	Aprendizaje automático y minería de datos	150	OB
1º	Sistemas multiagente y percepción computacional	150	OB
1º	Computación bioinspirada	150	OB

Mtra.Tere Guevara Navarro
Rectora

tech universidad tecnológica

*Apostilla de La Haya. En caso de que el alumno solicite que su título en papel recabe la Apostilla de La Haya, TECH EDUCATION realizará las gestiones oportunas para su obtención, con un coste adicional.

tech universidad tecnológica

Máster Título Propio

Inteligencia Artificial e Ingeniería del Conocimiento

Modalidad: **Online**

Duración: **12 meses**

Titulación: **TECH Universidad Tecnológica**

Horas lectivas: **1.500 h.**

Máster Título Propio

Inteligencia Artificial e Ingeniería del Conocimiento



tech universidad tecnológica