Professional Master's Degree

Artificial Intelligence and Knowledge Engineering

tech technological university

tech technological university

Professional Master's Degree

Artificial Intelligence and Knowledge Engineering

- Modality: **online**
- Duration: **12 months**
- Certificate: **TECH Technological University**
- Dedication: **16h/week**
- Schedule: **at your own pace**
- Exams: **online**

Website: **www.techtitute.com/pk/engineering/professional-master-degree/master-artificial-intelligence-knowledge-engineering**

Index

01

Introduction

This program is designed to immerse engineering professionals in the exciting world of Artificial Intelligence and Knowledge Engineering developments. Through this program, students will be able to take a solid and solvent step in this field, achieving the personal and professional skills necessary to practice as an expert in the field. A complete and effective program that will propel you to a higher skill level.

Become one of the most in-demand professionals of today. Get qualified *in a Professional Master's Degree in Artificial Intelligence and Knowledge Engineering with this comprehensive online program"*

Developments based on Artificial Intelligence have already reached numerous developments in the field of engineering. From the automation of numerous procedures in industry and companies, to its own process control. This means that engineering professionals need to understand and master the operation of these complex techniques.

This essential knowledge also becomes the first step to access the development capacity of this type of technology.

Throughout this program, a real working scenario is offered in order to be able to assess the convenience of its application in this project, evaluating its real indications, way of development and possible result expectations

Through experience, students will learn how to develop the knowledge required to advance in this field of work. This course, which necessarily requires experience, is coordinated through e-learning and practical teaching, offering a unique option to give your professional profile the boost you are looking for.

Join the elite, with this highly effective educational training and open new paths to help you advance in your professional progress"

This **Professional Master's Degree in Artificial Intelligence and Knowledge Engineering** contains the most complete and up-to-date educational program on the market. Its most notable features are:

- The latest technology in online teaching software
- A highly visual teaching system, supported by graphic and schematic contents that are easy to assimilate and understand
- Practical cases presented by practising experts
- State-of-the-art interactive video systems
- Teaching supported by telepractice
- Continuous updating and recycling systems
- Autonomous learning: full compatibility with other occupations
- Practical exercises for self-evaluation and learning verification
- Support groups and educational synergies: questions to the expert, debate and knowledge forums
- Communication with the teacher and individual reflection work
- Content that is accessible from any fixed or portable device with an Internet connection
- Supplementary documentation databases are permanently available, even after the program

A Professional Master's Degree that will enable you to work in all areas of Artificial Intelligence and Knowledge Engineering with the solvency of a high-level professional"

Our teaching staff is made up of professionals from different fields related to this specialty. This way, the intended educational updating objective is achieved. A multidisciplinary team of professionals trained and experienced in different fields, who will develop the theoretical knowledge in an efficient way, but above all, they will bring their practical knowledge from their own experience to the course: one of the differential qualities of this program.

This mastery of the subject matter is complemented by the effectiveness of the methodological design. Developed by a multidisciplinary team of e-learning experts, it integrates the latest advances in educational technology. In this way, you will be able to study with a range of comfortable and versatile multimedia tools that will give you the operability you need in your training.

The design of this program is based on Problem-Based Learning: An approach that conceives learning as a highly practical process. To achieve this remotely, telepractice will be used. With the help of an innovative, interactive video system and Learning from an Expert, students will be able to acquire the knowledge as if they were dealing with the case in real time. A concept that will make it possible to integrate and fix learning in a more realistic and permanent way.

Join the elite, with this highly effective educational training and open new paths to help you advance in your professional progress.

With the experience of active professionals, experts in Artificial Intelligence and Knowledge Engineering.

02

Objectives

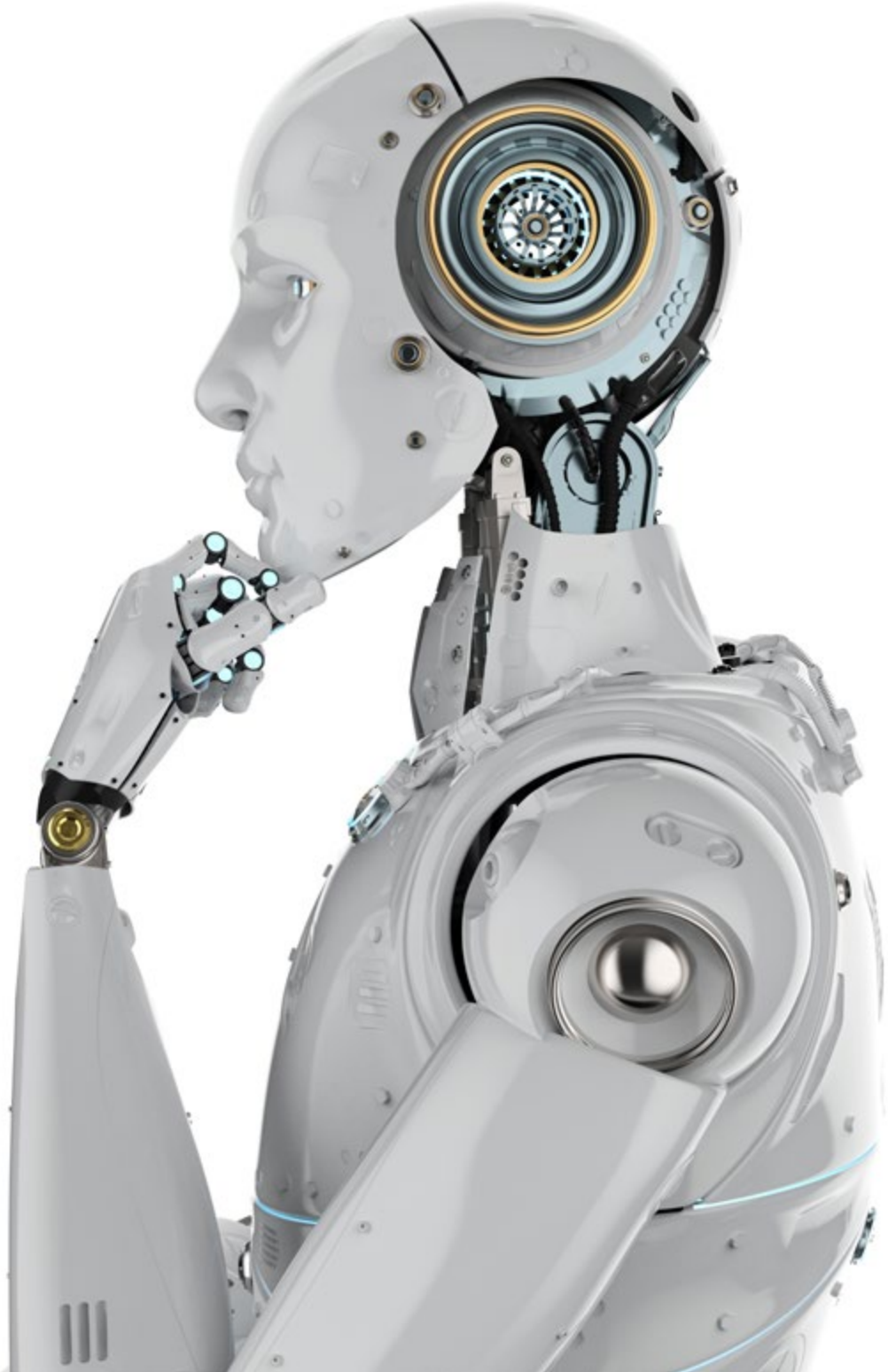
The objective is to train highly qualified professionals for work experience. An objective that is complemented in a global manner by promoting human development that lays the foundations for a better society. This objective materializes by helping professionals reach a much higher level of expertise and control. A goal that, in just a few months, students will be able to achieve with a highly intensive and precise course.

If your objective is to broaden your skills set to include new paths of success and development: this is the program for you: a training that aspires to excellence"

General Objectives

- Obtain scientific and technological knowledge to practice Mechanical Engineering
- Obtain extensive knowledge in the field of computing
- Obtain extensive knowledge in the field of structure of computing
- Acquire the necessary knowledge of software engineering
- Review the mathematical, statistical and physical foundations essential for this subject

A path to achieve training and professional growth that will propel you towards a greater level of competitiveness in the employment market"



Specific Objectives

Module 1. Programming Fundamentals

- Understand the basic structure of computers, software and the general-purpose programming languages
- Learn how to design and interpret algorithms, the necessary basis for developing computer programs
- Understand the essential elements of a computer program, such as the different data types, operators, expressions, statements, I/O and control statements

Module 2. Data Structure

- Understand the different data structures available in general purpose programming languages, both static and dynamic, as well as acquire essential knowledge in file handling
- Understand abstract data types, linear data structure types, simple and complex hierarchical data structures, and their implementation in C++
- Gain an understanding of the operation of advanced data structures, different to the usual ones
- Know the theory and practice related to the use of priority mounds and queues
- Learn how hash tables work as abstract data types and functions

Module 3. Algorithms and Complexity

- Understand the different software testing techniques and the importance of generating good documentation together with good source code
- Learn the basics of the C++ programming language, one of the most widely used programming languages in the world
- Learn the basics of programming in the C++ language, including classes, variables, conditional expressions, and objects
- Understanding graph theory, as well as advanced graph algorithms and concepts

Module 4. Advanced Algorithm Design

- Learn the main strategies for algorithm design, as well as the different methods and measures for algorithm computation
- Know the main sorting algorithms used in software development
- Understand the operation of different algorithms with trees, heaps and graphs
- Understand the operation of greedy algorithms, their strategy and examples of their use in the main known problems We will also learn about the use of greedy algorithms on graphs
- Learn the main strategies of minimum path finding, with the approach of essential problems of the field and algorithms for their resolution
- Understand the backtracking technique and its main uses, as well as other alternative techniques
- Delve into advanced algorithm design, analyzing recursive and divide and conquer algorithms, as well as performing amortized analysis
- Understand dynamic programming concepts and algorithms for NP-problems
- Understand the operation of combinatorial optimization, as well as the different randomization algorithms and parallel algorithms
- Know and understand the operation of the different local and candidate search methods

Module 5. Logic in Computer Science

- Learn the mechanisms of formal program and iterative program verification, including first-order logic and Hoare's formal system
- Learn the operation of some of the main numerical methods such as the bisection method, the Newton Raphson method and the secant method
- Learn the basics of logic in computer science, what it is used for and its justification for use
- Know the different strategies of formalization and deduction in propositional logic, including natural reasoning, axiomatic and natural deduction, as well as the primitive rules of propositional calculus
- Acquire advanced knowledge in propositional logic, delving into its semantics and the main applications of this logic, such as logic circuits
- Understand predicate logic for both natural predicate deduction calculus and formalization and deduction strategies for predicate logic
- Understand the fundamentals of natural language and its deductive mechanism

Module 6. Artificial Intelligence and Knowledge Engineering

- Lay the foundations of artificial intelligence and knowledge engineering, making a brief tour through the history of artificial intelligence up to the present day
- Understand the essential concepts of search in artificial intelligence, both informed and uninformed search
- Understand how artificial intelligence works in games
- Learn the fundamental concepts of neural networks and the use of genetic algorithms
- Acquire the appropriate mechanisms to represent knowledge, especially taking into account the semantic web
- Understand the operation of expert systems and decision support systems

Module 7. Intelligent Systems

- Learn all the concepts related to agent theory and agent architecture along with its reasoning process
- Assimilate the theory and practice behind the concepts of information and knowledge, as well as the different ways of representing knowledge
- Understand the theory related to ontologies, as well as learn languages and software for ontology creation
- Learn different models of knowledge representation, such as vocabularies, taxonomies, thesauri and mind maps, among others
- Understand the functioning of semantic reasoners, knowledge-based systems and expert systems
- Understand how the semantic web works, its current and future status, as well as applications based on semantic web

Module 8. Machine Learning and Data Mining

- Introduce knowledge discovery processes and basic concepts of machine learning
- Learn methods of data exploration and preprocessing, as well as different algorithms based on decision trees
- Understand the operation of Bayesian methods and regression and continuous response methods
- Understand the different classification rules and the evaluation of classifiers, learning to use confusion matrices and numerical evaluation, the Kappa statistic and the ROC curve
- Acquire essential knowledge related to text mining and natural language processing (NLP) and clustering
- Gain an in-depth understanding of neural networks, from simple to recurrent

Module 9. Multiagent Systems and Computational Perception

- Understand the basic and advanced concepts related to agents and multi-agent systems
- Study the FIPA agent standard, taking into account inter-agent communication, agent management, architecture and other issues
- Further study the JADE platform (Java Agent Development Framework), learning to program in it both basic and advanced concepts, including communication and agent discovery topics
- Lay the foundations of natural language processing, such as automatic speech recognition and computational linguistics
- Understand, in-depth, computer vision, digital image analysis, image transformation and segmentation

Module 10. Bio-Inspired Computing

- Introduce the concept of bio-inspired computing, as well as to understand the functioning of different types of social adaptation and genetic algorithms
- Further study the different models of evolutionary computation, understanding their strategies, programming, algorithms and models based on estimation of distributions
- Understand the main space exploration-exploitation strategies for genetic algorithms
- Understand the functioning of evolutionary programming applied to learning problems and multi-objective problems
- Learn the essential concepts related to neural networks and understand the operation of real use cases applied to areas as diverse as medical research, economics and computer vision

03

Skills

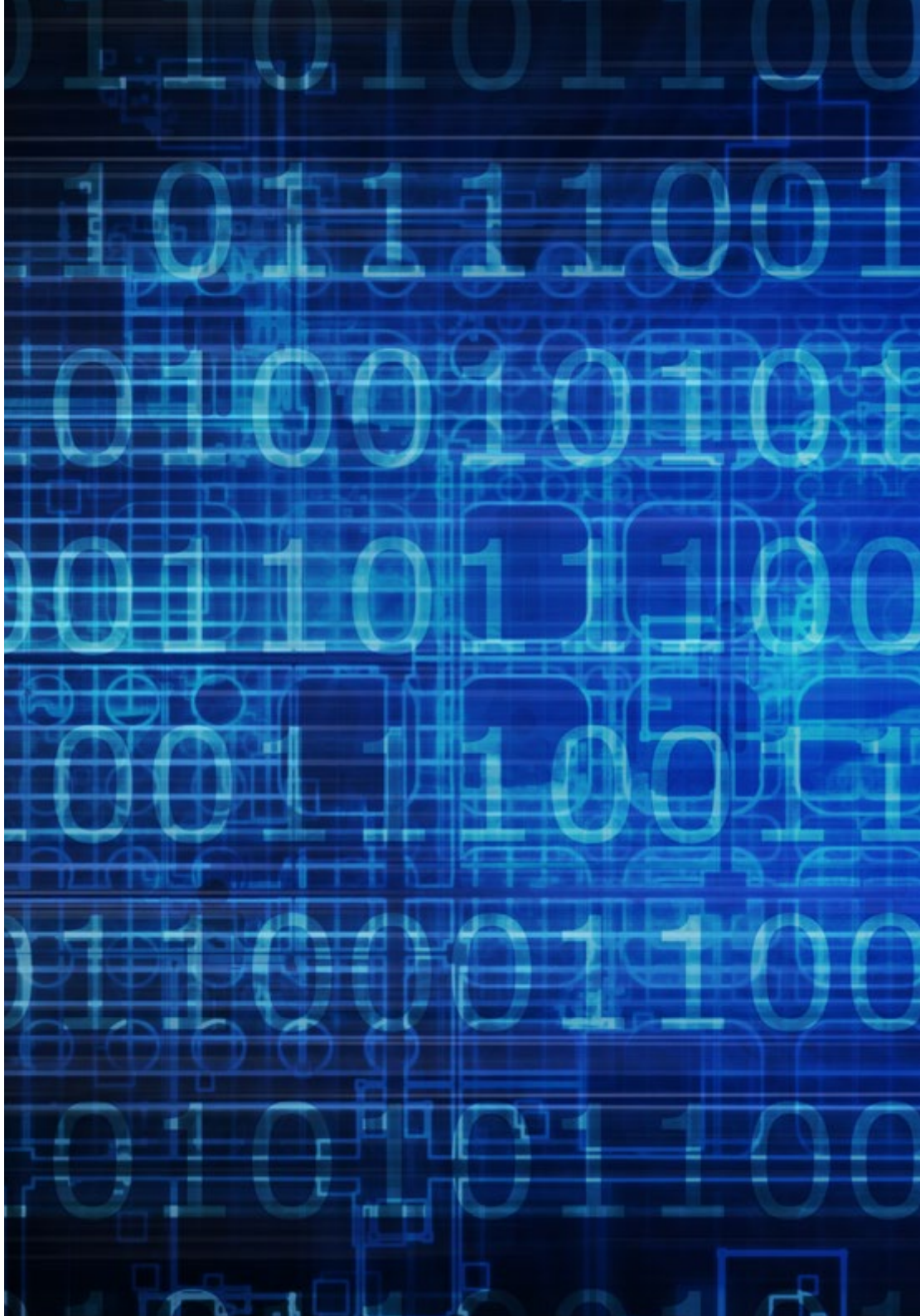
This Professional Master's Degree in Artificial Intelligence and Knowledge Engineering has been created as a high-quality preparation tool for professionals. This intensive course will allow you to work in all the fields related to this industry with the confidence of an expert.

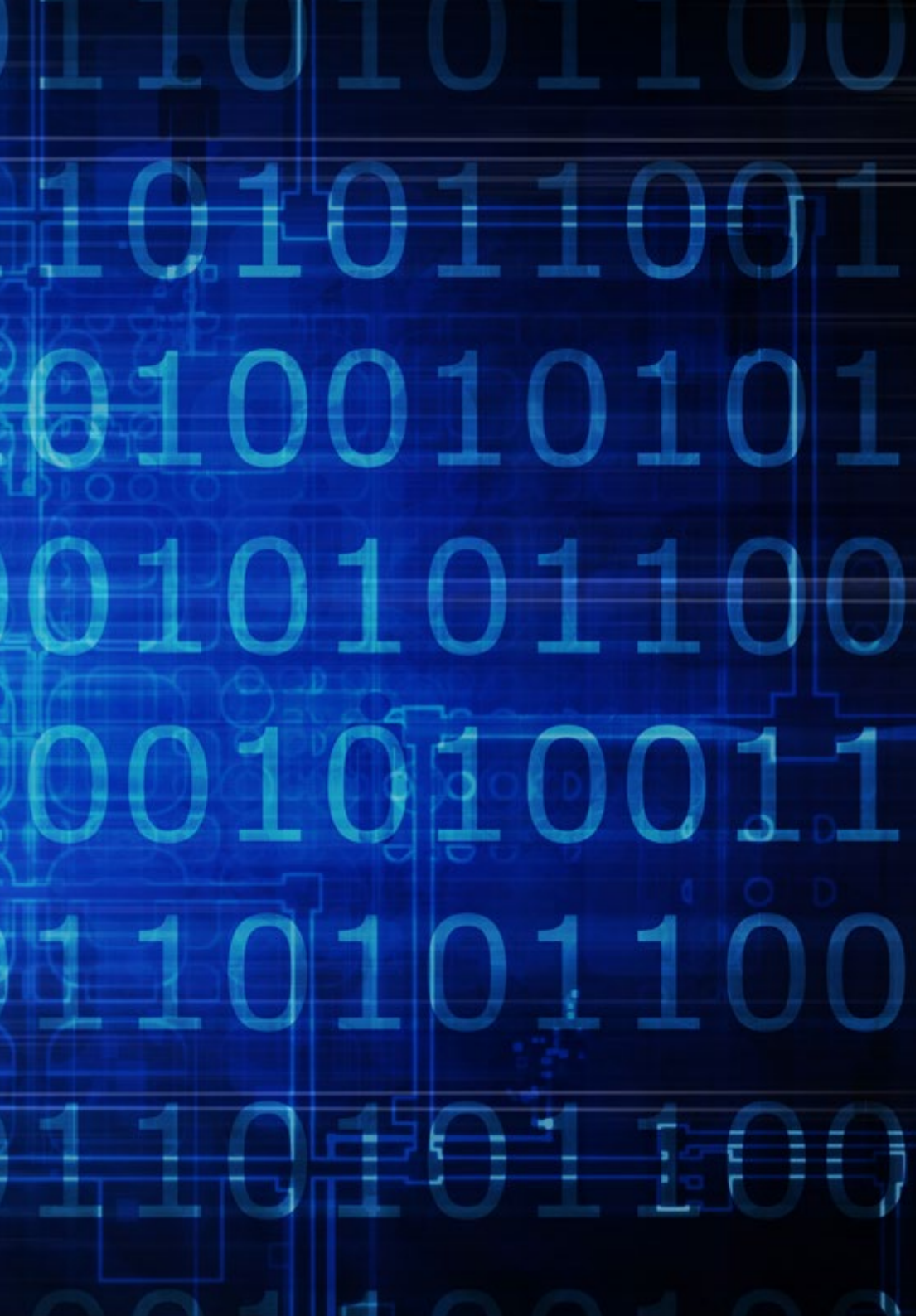
This Professional Master's Degree in Artificial Intelligence and Knowledge Engineering will provide you with the essential personal and professional skills to play an adequate role in any professional situation in this field of intervention"

General Skill

•Acquire the necessary skills for the professional practice of computer engineering with the knowledge of all the necessary factors to perform with quality and solvency

A unique, key, and decisive experience to boost your professional development"





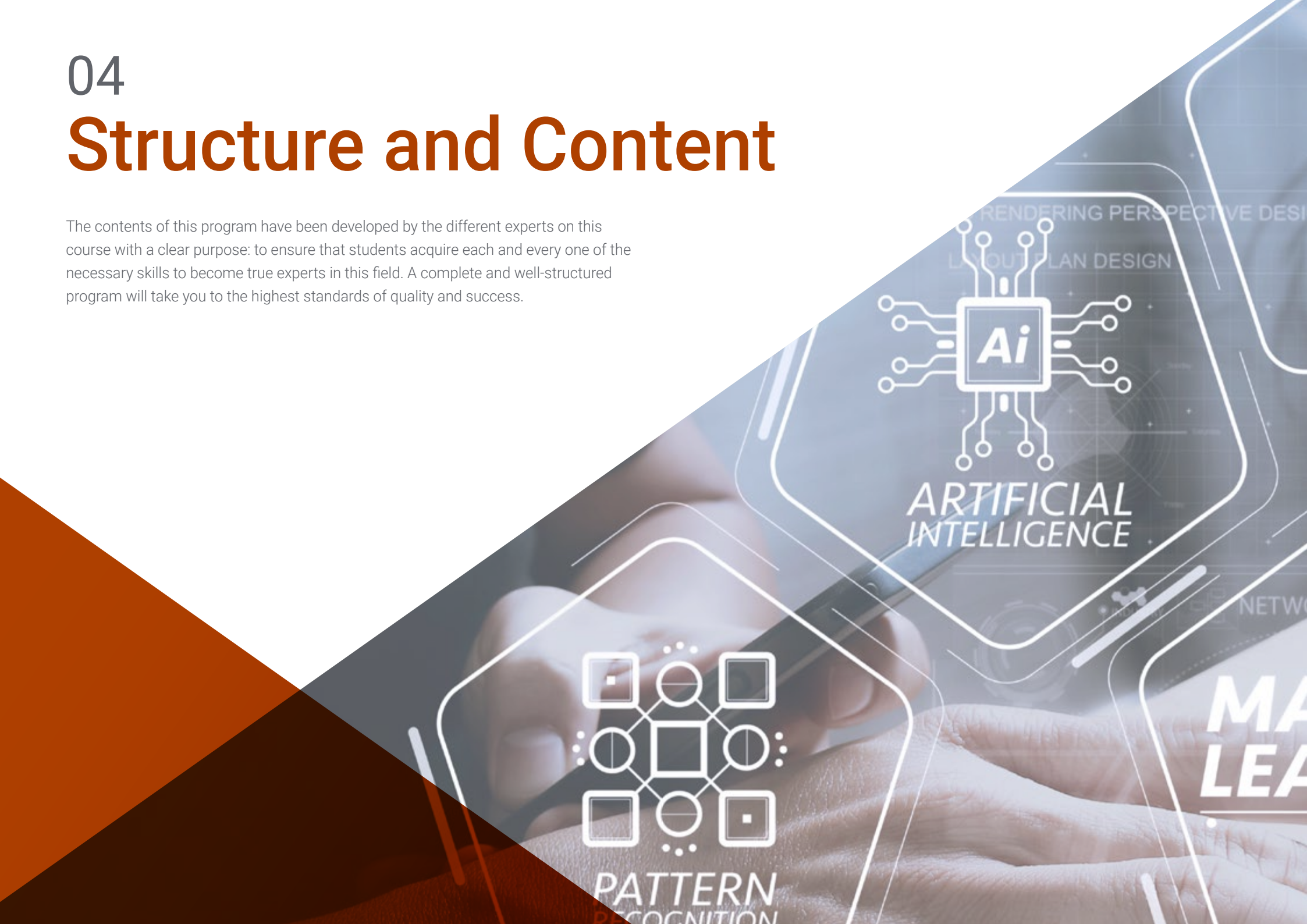
Specific Skills

- Develop programming in the area of artificial intelligence, taking into account all factors of artificial intelligence development
- Have a good knowledge of the data structure in C++ programming
- Design basic and advanced algorithms
- Understand computational logic and apply it to project design
- Understand artificial intelligence, its uses and developments and implement your own projects
- Know what they are, how they work and how to work with intelligent systems
- Master the basic concepts of machine learning
- Understand JADE, FIPA, Computer Vision and other multi-agent systems
- Understand bio-inspired computing algorithms and utilization strategies

04

Structure and Content

The contents of this program have been developed by the different experts on this course with a clear purpose: to ensure that students acquire each and every one of the necessary skills to become true experts in this field. A complete and well-structured program will take you to the highest standards of quality and success.



A comprehensive teaching program, structured in well-developed teaching units, oriented towards learning that is compatible with your personal and professional life"

Module 1. Programming Fundamentals

1.1. Introduction to Programming
- 1.1.1. Basic Computer Structure
- 1.1.2. Software
- 1.1.3. Programming Languages
- 1.1.4. Computer Application Life Cycle

1.2. Algorithm Design
- 1.2.1. Problem Solving
- 1.2.2. Descriptive Techniques
- 1.2.3. Algorithm Elements and Structure

1.3. Program Elements
- 1.3.1. C++ Origin and Features
- 1.3.2. Development Environment
- 1.3.3. Concept of Program
- 1.3.4. Types of Fundamental Data
- 1.3.5. Operators
- 1.3.6. Expressions
- 1.3.7. Statements
- 1.3.8. Data Input and Output

1.4. Control Statements
- 1.4.1. Statements
- 1.4.2. Branches
- 1.4.3. Loops

1.5. Abstraction and Modularity: Functions
- 1.5.1. Modular Design
- 1.5.2. Function and Utility Concepts
- 1.5.3. Definition of Function
- 1.5.4. Execution Flow in a Function Call
- 1.5.5. Function Prototypes
- 1.5.6. Results Return
- 1.5.7. Function Calls: Parameters
- 1.5.8. Parameter Passing by Reference and Value
- 1.5.9. Scope Identifier

1.6. Statistical Data Structures
- 1.6.1. Arrays
- 1.6.2. Matrices Polyhedra
- 1.6.3. Searching and Sorting
- 1.6.4. Chaining: I/O Functions for Chains
- 1.6.5. Structures: Unions
- 1.6.6. New Types of Data

1.7. Dynamic Data Structures: Pointers
- 1.7.1. Concept. Definition of Pointer
- 1.7.2. Pointer Operators and Operations
- 1.7.3. Pointer Arrays
- 1.7.4. Pointers and Arrays
- 1.7.5. Chains of Pointers
- 1.7.6. Structure Pointers
- 1.7.7. Multiple Indirection
- 1.7.8. Function Pointers
- 1.7.9. Function, Structure and Array Passing as Function Parameters

1.8. Files
- 1.8.1. Basic Concepts
- 1.8.2. File Operations
- 1.8.3. Types of Files
- 1.8.4. File Organization
- 1.8.5. Introduction to C++ Files
- 1.8.6. Managing Files

1.9. Recursion
- 1.9.1. Definition of Recursion
- 1.9.2. Types of Recursion
- 1.9.3. Advantages and Disadvantages
- 1.9.4. Considerations
- 1.9.5. Recursive-Iterative Conversion
- 1.9.6. Recursion Stack

1.10. Testing and Documentation
- 1.10.1. Program Testing
- 1.10.2. White Box Testing
- 1.10.3. Black Box Testing
- 1.10.4. Testing Tools
- 1.10.5. Program Documentation

Module 2. Data Structure

2.1. Introduction to C++ Programming
- 2.1.1. Classes, Constructors, Methods and Attributes
- 2.1.2. Variables:
- 2.1.3. Conditional Expressions and Loops
- 2.1.4. Objects

2.2. Abstract Data Types (ADT)
- 2.2.1. Types of Data
- 2.2.2. Basic Structures and TADs
- 2.2.3. Vectors and Arrays

2.3. Linear Data Structures
- 2.3.1. TAD List Definition
- 2.3.2. Linked and Double-Linked Lists
- 2.3.3. Ordered Lists
- 2.3.4. Lists in C++
- 2.3.5. TAD Stack
- 2.3.6. TAD Queue
- 2.3.7. Stacks and Queues in C++

2.4. Hierarchical Data Structures
- 2.4.1. TAD Tree
- 2.4.2. Tree Traversal
- 2.4.3. N-ary Trees
- 2.4.4. Binary Trees
- 2.4.5. Binary Search Trees

2.5. Hierarchical Data Structures: Complex Trees
- 2.5.1. Perfectly Balanced Trees or Trees of Minimal Height
- 2.5.2. Multi-path Trees
- 2.5.3. Bibliographical References

2.6. Priority Heaps and Queues
- 2.6.1. TAD Heaps
- 2.6.2. TAD Priority Queue

2.7. Hash Tables
- 2.7.1. TAD Hash Tables
- 2.7.2. Hash Functions
- 2.7.3. Hash Function in Hash Tables
- 2.7.4. Redispersion
- 2.7.5. Open Hash Tables

2.8. Graphs
- 2.8.1. TAD Graphs
- 2.8.2. Types of Graphs
- 2.8.3. Graphic Representation and Basic Operations
- 2.8.4. Graph Design

2.9. Algorithms and Advanced Graph Concepts
- 2.9.1. Problems with Graphs
- 2.9.2. Path Algorithms
- 2.9.3. Search or Path Algorithms
- 2.9.4. Other Algorithms

2.10. Other Data Structures
- 2.10.1. Sets
- 2.10.2. Parallel Arrays
- 2.10.3. Table of Symbols
- 2.10.4. Tries

Module 3. Algorithms and Complexity

3.1. Introduction to Algorithm Design Strategies
- 3.1.1. Recursion
- 3.1.2. Divide and Conquer
- 3.1.3. Other Strategies

3.2. Algorithm Efficiency and Analysis
- 3.2.1. Efficiency Measures
- 3.2.2. Measuring Entry Size
- 3.2.3. Measuring Execution Time
- 3.2.4. Worst, Best and Average Case
- 3.2.5. Asymptotic Notation
- 3.2.6. Mathematical Analysis Criteria for Non-Recursive Algorithms
- 3.2.7. Mathematical Analysis for Recursive Algorithms
- 3.2.8. Empirical Analysis for Algorithms

3.3. Sorting Algorithms
- 3.3.1. Concept of Sorting
- 3.3.2. Bubble Sorting
- 3.3.3. Selection Sorting
- 3.3.4. Insertion Sorting
- 3.3.5. Mixed Sorting (merge_sort)
- 3.3.6. Quick Sorting (quick_sort)

3.4. Tree Algorithms
- 3.4.1. Concept of Trees
- 3.4.2. Binary Trees
- 3.4.3. Tree Traversal
- 3.4.4. Representing Expressions
- 3.4.5. Sorted Binary Trees
- 3.4.6. Balanced Binary Trees

3.5. Algorithms Using Heaps
- 3.5.1. Heaps
- 3.5.2. The Heapsort Algorithm
- 3.5.3. Priority Queues

3.6. Graph Algorithms
- 3.6.1. Representation
- 3.6.2. Width Tree Traversal
- 3.6.3. Depth Tree Traversal
- 3.6.4. Topological Sorting

3.7. Greedy Algorithms
- 3.7.1. Greedy Strategy
- 3.7.2. Greedy Strategy Elements
- 3.7.3. Currency Exchange
- 3.7.4. Traveler's Problem
- 3.7.5. Backpack Problem

3.8. Minimal Pathways Search
- 3.8.1. Shortest Path Problem
- 3.8.2. Cycles and Negative Arcs
- 3.8.3. Dijkstra's Algorithm

3.9. Greedy Algorithms on Graphs
- 3.9.1. Minimum Spanning Tree
- 3.9.2. Prim's Algorithm
- 3.9.3. Kruskal's Algorithm
- 3.9.4. Complexity Analysis

3.10. Backtracking
- 3.10.1. Backtracking
- 3.10.2. Alternative Techniques

Module 4. Advanced Algorithm Design

4.1. Analysis of Recursive and Divide-and-Conquer Algorithms
- 4.1.1. Posing and Solving of Homogeneous and Non-Homogeneous Recurrence Equations
- 4.1.2. Divide and Conquer Strategy Overview

4.2. Amortized Analysis
- 4.2.1. Aggregate Analysis
- 4.2.2. Accounting Method
- 4.2.3. Potential Method

4.3. Dynamic Programming and Algorithms for NP-Problems
- 4.3.1. Characteristics of Dynamic Programming
- 4.3.2. Backtracking: Backtracking
- 4.3.3. Branching and Pruning

4.4. Combinatorial Optimization
- 4.4.1. Problem Representation
- 4.4.2. 1D Optimization

4.5. Randomization Algorithms
- 4.5.1. Examples of Randomization Algorithms
- 4.5.2. The Buffon Theorem
- 4.5.3. Monte Carlo Algorithm
- 4.5.4. Las Vegas Algorithm

4.6. Local and Candidate Searches
- 4.6.1. Gradient Ascent
- 4.6.2. Hill Climbing
- 4.6.3. Simulated Annealing
- 4.6.4. Taboo Search
- 4.6.5. Candidate Searches

4.7. Formal Program Verification
- 4.7.1. Specification of Functional Abstractions
- 4.7.2. Language of First-Order Logic
- 4.7.3. Hoare's Formal System

4.8. Iterative Program Verification
- 4.8.1. Rules of Hoare's Formal System
- 4.8.2. Concept of Invariant Iterations

4.9. Numeric Methods
- 4.9.1. Bisection Methods
- 4.9.2. Newton Raphson Method
- 4.9.3. Secant Methods

4.10. Parallel Algorithms
- 4.10.1. Parallel Binary Operations
- 4.10.2. Parallel Operations with Graphs
- 4.10.3. Parallelism in Divide-and-Conquer
- 4.10.4. Parallelism in Dynamic Programming

Module 5. Logic in Computer Science

5.1. Justification of Logic
- 5.1.1. Object of the Study of Logic
- 5.1.2. What is Logic For?
- 5.1.3. Components and Types of Reasoning
- 5.1.4. Components of Logic Calculations
- 5.1.5. Semantics
- 5.1.6. Justification of the Existence of Logic
- 5.1.7. How to Check that Logic is Adequate

5.2. Calculation of Natural Deduction from Statements
- 5.2.1. Formal Language
- 5.2.2. Deductive Mechanism

5.3. Formalization and Deduction Strategies for Propositional Logic
- 5.3.1. Formalization Strategies
- 5.3.2. Natural Reasoning
- 5.3.3. Laws and Rules
- 5.3.4. Axiomatic Natural Deduction
- 5.3.5. Calculating Natural Deduction
- 5.3.6. Primitive Rules of Propositional Calculus

5.4. Semantics of Propositional Logic
- 5.4.1. Truth Tables
- 5.4.2. Equivalence
- 5.4.3. Tautologies and Contradictions
- 5.4.4. Validation of Propositional Sentences
- 5.4.5. Validation Using Truth Tables
- 5.4.6. Validation Using Semantic Trees
- 5.4.7. Validation by Refutation

5.5. Applications of Propositional Logic: Logic Circuits
- 5.5.1. Basic Doors
- 5.5.2. Circuits
- 5.5.3. Mathematical Circuit Models
- 5.5.4. Minimization
- 5.5.5. The Second Canonical and Minimum Form in Product of Additions
- 5.5.6. Other Doors

5.6. Natural Predicate Deduction Calculus
- 5.6.1. Formal Language
- 5.6.2. Deductive Mechanism

5.7. Formalization Strategies for Predicate Logic
- 5.7.1. Introduction to Formalization in Predicate Logic
- 5.7.2. Formalization Strategies with Quantifiers

5.8. Deduction Strategies for Predicate Logic
- 5.8.1. Reason for Omission
- 5.8.2. Presentation of New Rules
- 5.8.3. Predicate logic as a Natural Deduction Calculus

5.9. Applications of Predicate Logic: Introduction to Logic Programming
- 5.9.1. Informal Presentation
- 5.9.2. Prolog Elements
- 5.9.3. Re-evaluation and Cutoff

5.10. Set Theory, Predicate Logic and its Semantics
- 5.10.1. Intuitive Set Theory
- 5.10.2. Introduction to Predicate Semantics

Module 6. Artificial Intelligence and Knowledge Engineering

6.1. Introduction to Artificial Intelligence and Knowledge Engineering
- 6.1.1. Brief History of Artificial Intelligence
- 6.1.2. Artificial Intelligence Today
- 6.1.3. Knowledge Engineering

6.2. Searching
- 6.2.1. Common Search Concepts
- 6.2.2. Non-informed Search
- 6.2.3. Informed Search

6.3. Boolean Satisfiability, Constraint Satisfiability and Automatic Planning
- 6.3.1. Boolean Satisfiability
- 6.3.2. Constraint Satisfaction Problems
- 6.3.3. Automatic Planning and PDDL
- 6.3.4. Planning as a Heuristic Search
- 6.3.5. Planning with SAT

6.4. Artificial Intelligence in Games
- 6.4.1. Game Theory
- 6.4.2. Minimax and Alpha-Beta Pruning
- 6.4.3. Simulation: Monte Carlo

6.5. Supervised and Unsupervised Learning
- 6.5.1. Introduction to Machine Learning
- 6.5.2. Classification
- 6.5.3. Regression
- 6.5.4. Validation of Results
- 6.5.5. Clustering

6.6. Neuron Networks
- 6.6.1. Biological Fundamentals
- 6.6.2. Computational Model
- 6.6.3. Supervised and Unsupervised Neural Networks
- 6.6.4. Simple Perceptron
- 6.6.5. Multilayer Perceptron

6.7. Genetic Algorithms
- 6.7.1. History
- 6.7.2. Biological Basis
- 6.7.3. Problem Coding
- 6.7.4. Initial Population Generation
- 6.7.5. Main Algorithm and Genetic Operators
- 6.7.6. Evaluation of Individuals: Fitness

6.8. Thesauri, Vocabularies, and Taxonomies
- 6.8.1. Vocabulary
- 6.8.2. Taxonomy
- 6.8.3. Thesauri
- 6.8.4. Ontologies

6.9. Knowledge Representation: Semantic Web
- 6.9.1. Semantic Web
- 6.9.2. Specifications RDF, RDFS and OWL
- 6.9.3. Inference/Reasoning
- 6.9.4. Linked Data

6.10. Expert Systems and DSS
- 6.10.1. Expert Systems
- 6.10.2. Decision Support Systems

Module 7. Intelligent Systems

7.1. Agents Theory
- 7.1.1. Concept History
- 7.1.2. Agent Definition
- 7.1.3. Agents in Artificial Intelligence
- 7.1.4. Agents in Software Engineering

7.2. Agent Architectures
- 7.2.1. Agents Reasoning Process
- 7.2.2. Reactive Agents
- 7.2.3. Deductive Agents
- 7.2.4. Hybrid Agents
- 7.2.5. Comparison

7.3. Information and Knowledge
- 7.3.1. Difference between Data, Information and Knowledge
- 7.3.2. Data Quality Assessment
- 7.3.3. Data Collection Methods
- 7.3.4. Information Acquisition Methods
- 7.3.5. Knowledge Acquisition Methods

7.4. Knowledge Representation
- 7.4.1. The Importance of Knowledge Representation
- 7.4.2. Definition of Knowledge Representation Through Their Roles
- 7.4.3. Knowledge Representation Features

7.5. Ontologies
- 7.5.1. Introduction to Metadata
- 7.5.2. Philosophical Concept of Ontology
- 7.5.3. Computing Concept of Ontology
- 7.5.4. Domain Ontologies and Higher-Level Ontologies
- 7.5.5. Building an Ontology

7.6. Ontology Languages and Ontology Creation Software
- 7.6.1. Triple RDF, Turtle and N3
- 7.6.2. RDF Schema
- 7.6.3. OWL
- 7.6.4. SPARQL
- 7.6.5. Introduction to Ontology Creation Tools
- 7.6.6. Installing and Using Protégé

7.7. Semantic Web
- 7.7.1. Current and Future Status of Semantic Web
- 7.7.2. Semantic Web Applications

7.8. Other Knowledge Representation Models
- 7.8.1. Vocabulary
- 7.8.2. Global Vision
- 7.8.3. Taxonomy
- 7.8.4. Thesauri
- 7.8.5. Folksonomy
- 7.8.6. Comparison
- 7.8.7. Mind Maps

7.9. Knowledge Representation Assessment and Integration
- 7.9.1. Zeroth-Order Logic
- 7.9.2. First-Order Logic
- 7.9.3. Description Logic
- 7.9.4. Relation between Different Types of Logic
- 7.9.5. Prolog: Programming Based on First-Order Logic

7.10. Semantic Reasoners, Knowledge-Based Systems and Expert Systems
- 7.10.1. Concept of Reasoner
- 7.10.2. Reasoner Applications
- 7.10.3. Knowledge-Based Systems
- 7.10.4. MYCIN: History of Expert Systems
- 7.10.5. Expert Systems Elements and Architecture
- 7.10.6. Creating Expert Systems

Module 8. Machine Learning and Data Mining

8.1. Introduction to knowledge discovery processes and basic concepts of machine learning
- 8.1.1. Key Concepts of Knowledge Discovery Processes
- 8.1.2. Historical Perspective of Knowledge Discovery Processes
- 8.1.3. Stages of Knowledge Discovery Processes
- 8.1.4. Key Concepts of Knowledge Discovery Processes
- 8.1.5. Characteristics of Good Machine Learning Models
- 8.1.6. Types of Machine Learning Data
- 8.1.7. Basic Learning Concepts
- 8.1.8. Basic Concepts of Unsupervised Learning

8.2. Data Exploration and Pre-Processing
- 8.2.1. Data Processing
- 8.2.2. Data Processing in the Data Analysis Flow
- 8.2.3. Types of Data
- 8.2.4. Data Transformation
- 8.2.5. Visualization and Exploration of Continuous Variables
- 8.2.6. Visualization and Exploration of Categorical Variables
- 8.2.7. Correlation Measures
- 8.2.8. Most Common Graphic Representations
- 8.2.9. Introduction to Multivariate Analysis and Dimensionality Reduction

8.3. Decision Trees
- 8.3.1. ID3 Algorithm
- 8.3.2. C4.5 Algorithm
- 8.3.3. Over-Training and Pruning
- 8.3.4. Analysis of Results

8.4. Evaluation of Classifiers
- 8.4.1. Confusion Matrices
- 8.4.2. Numerical Evaluation Matrices
- 8.4.3. Kappa Statistics
- 8.4.4. ROC Curves

8.5. Classification Rules
- 8.5.1. Rule Evaluation Measures
- 8.5.2. Introduction to Graphic Representation
- 8.5.3. Sequential Overlay Algorithm

8.6. Neural Networks
- 8.6.1. Basic Concepts
- 8.6.2. Simple Neural Networks
- 8.6.3. Backpropagation Algorithm
- 8.6.4. Introduction to Recurrent Neural Networks

8.7. Bayesian Methods
- 8.7.1. Basic Concepts of Probability
- 8.7.2. Bayes' Theorem
- 8.7.3. Naive Bayes
- 8.7.4. Introduction to Bayesian Networks

8.8. Regression and Continuous Response Models
- 8.8.1. Simple Linear Regression
- 8.8.2. Multiple Linear Regression
- 8.8.3. Logistic Regression
- 8.8.4. Regression Trees
- 8.8.5. Introduction to Support Vector Machines (SVM)
- 8.8.6. Goodness-of-fit Measures

8.9. Clustering
- 8.9.1. Basic Concepts
- 8.9.2. Hierarchical Clustering
- 8.9.3. Probabilistic Methods
- 8.9.4. EM Algorithm
- 8.9.5. B-Cubed Method
- 8.9.6. Implicit Methods

8.10. Text Mining and Natural Language Processing (NLP)
- 8.10.1. Basic Concepts
- 8.10.2. Corpus Creation
- 8.10.3. Descriptive Analysis
- 8.10.4. Introduction to Feelings Analysis

Module 9. Multiagent Systems and Computational Perception

9.1. Agent and Multiagent systems
- 9.1.1. Agent Concept
- 9.1.2. Architecture
- 9.1.3. Communication and Coordination
- 9.1.4. Programming Languages and Tools
- 9.1.5. Agent Applications
- 9.1.6. FIPA

9.2. The Agent Standard: FIPA
- 9.2.1. Agent Communication
- 9.2.2. Agent Management
- 9.2.3. Abstract Architecture
- 9.2.4. Other Specifications

9.3. JADE Platform
- 9.3.1. JADE Software Agents
- 9.3.2. Architecture
- 9.3.3. Installation and Execution
- 9.3.4. JADE Packages

9.4. Basic JADE Programming
- 9.4.1. Management Consoles
- 9.4.2. Basic Agent Creation

9.5. Advanced JADE Programming
- 9.5.1. Advanced Agent Creation
- 9.5.2. Agent Communication
- 9.5.3. Agent Discovery

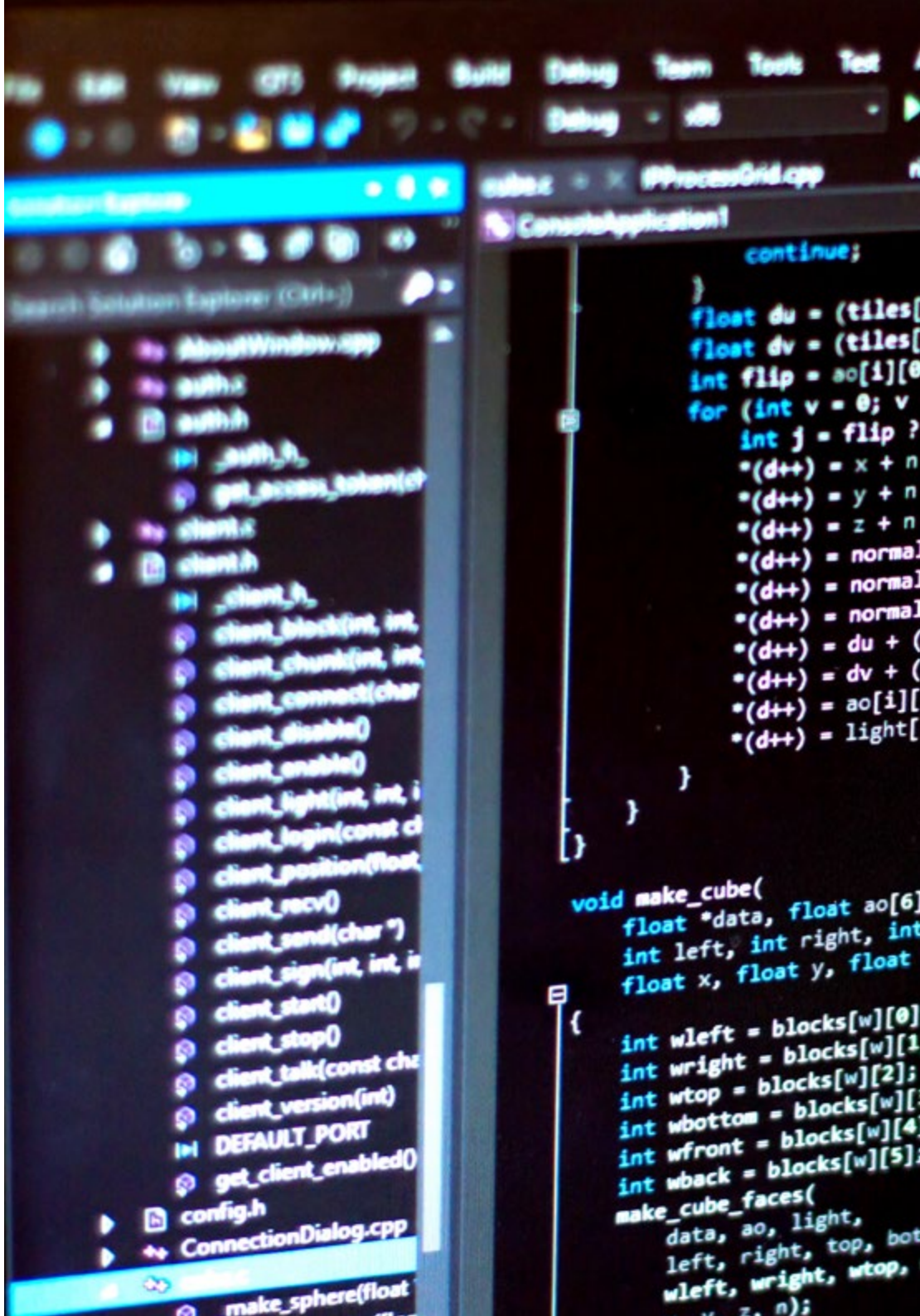
9.6. Artificial Vision
- 9.6.1. Digital Image Processing and Analysis
- 9.6.2. Image Analysis and Computer Vision
- 9.6.3. Image Processing and Human Vision
- 9.6.4. Image Capture Systems
- 9.6.5. Image and Perception Training

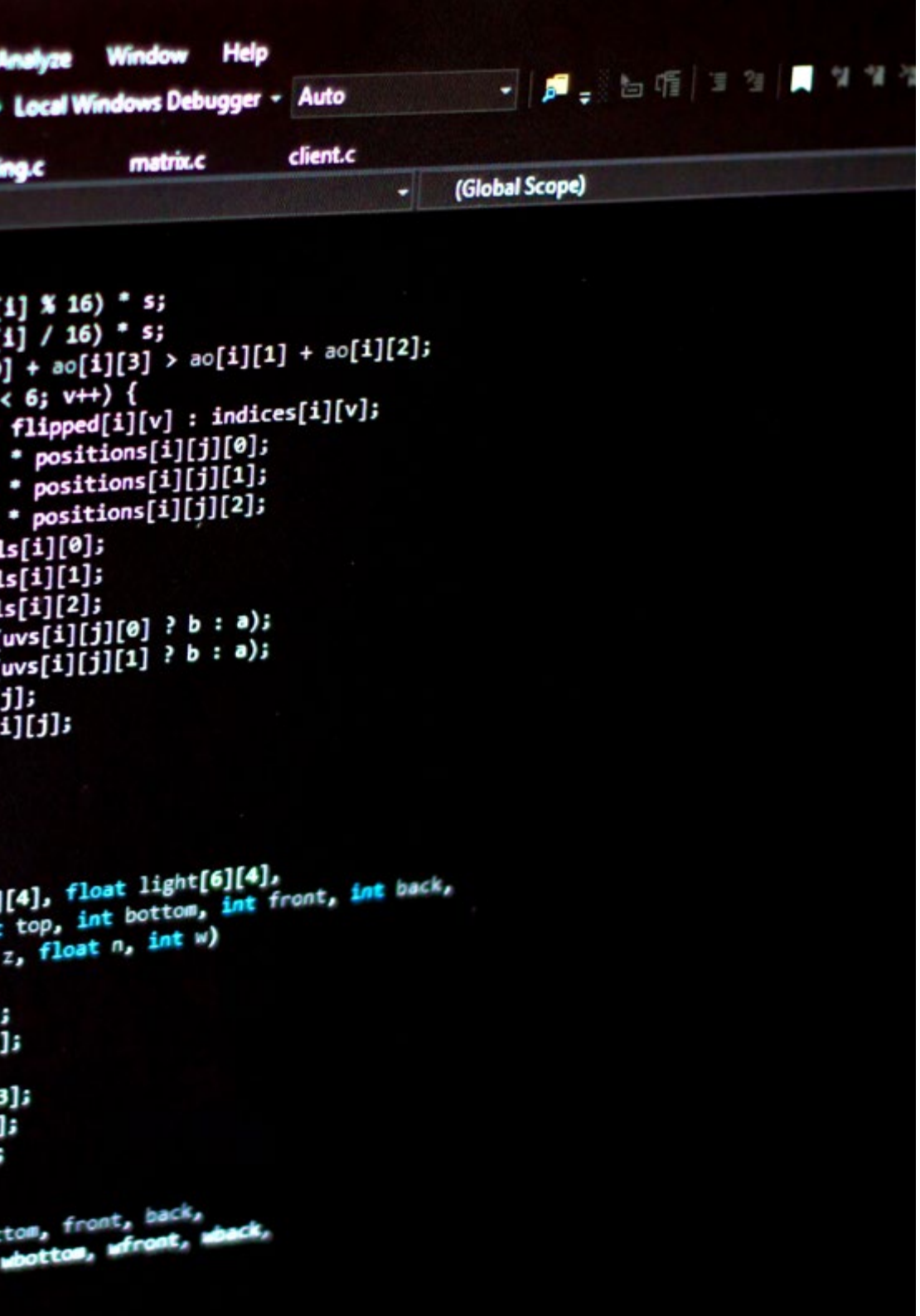
9.7. Digital Image Analysis
- 9.7.1. Stages of the Image Analysis Process
- 9.7.2. Pre-Processing
- 9.7.3. Basic Operations
- 9.7.4. Spatial Filtering

9.8. Digital Image Transformation and Image Segmentation
- 9.8.1. Fourier Transforms
- 9.8.2. Frequency Filtering
- 9.8.3. Basic Concepts
- 9.8.4. Thresholding
- 9.8.5. Contour Detection

9.9. Shape Recognition
- 9.9.1. Feature Extraction
- 9.9.2. Classification Algorithms

9.10. Processing Natural Language
- 9.10.1. Speech Recognition
- 9.10.2. Computational Linguistics





Module 10. Bio-Inspired Computing

10.1. Introduction to Bio-Inspired Computing
- 10.1.1. Introduction to Bio-Inspired Computing

10.2. Social Adaptation Algorithms
- 10.2.1. Bio-inspired Computing Based on Ant Colonies
- 10.2.2. Variants of Ant Colony Algorithms
- 10.2.3. Particle Cloud Computing

10.3. Genetic Algorithms
- 10.3.1. General Structure
- 10.3.2. Implementations of the Main Operators

10.4. Space Exploration-Exploitation Strategies for Genetic Algorithms
- 10.4.1. CHC Algorithm
- 10.4.2. Multimodal Problems

10.5. Models of Evolutionary Computation (I)
- 10.5.1. Evolutionary Strategies
- 10.5.2. Evolutionary Programming
- 10.5.3. Algorithms Based on Differential Evolution

10.6. Models of Evolutionary Computation (II)
- 10.6.1. Evolution Models Based on Estimation of Distributions (EDA)
- 10.6.2. Genetic Programming

10.7. Developmental Programming Applied to Learning Disabilities
- 10.7.1. Rules Based Learning
- 10.7.2. Evolutionary Methods in Instance Selection Problems

10.8. Multi-Objective Problems
- 10.8.1. Concept of Dominance
- 10.8.2. Application of Evolutionary Algorithms to Multiobjective Problems

10.9. Neural Networks (I)
- 10.9.1. Introduction to Neural Networks
- 10.9.2. Practical Example Using Neural Networks

10.10. Neural Networks (II)
- 10.10.1. Case Studies of Neural Networks in Medical Research
- 10.10.2. Case Studies of Neural Networks in Economy
- 10.10.3. Case Studies of Neural Networks in Artificial Vision

05

Methodology

This academic program offers students a different way of learning. Our methodology uses a cyclical learning approach: **Relearning.**

This teaching system is used, for example, in the most prestigious medical schools in the world, and major publications such as the **New England Journal of Medicine** have considered it to be one of the most effective.

Discover Relearning, a system that abandons conventional linear learning, to take you through cyclical teaching systems: a way of learning that has proven to be extremely effective, especially in subjects that require memorization"

Case Study to contextualize all content

Our program offers a revolutionary approach to developing skills and knowledge. Our goal is to strengthen skills in a changing, competitive, and highly demanding environment.

At TECH, you will experience a learning methodology that is shaking the foundations of traditional universities around the world"

You will have access to a learning system based on repetition, with natural and progressive teaching throughout the entire syllabus.

The student will learn to solve complex situations in real business environments through collaborative activities and real cases.

A learning method that is different and innovative

This TECH program is an intensive educational program, created from scratch, which presents the most demanding challenges and decisions in this field, both nationally and internationally. This methodology promotes personal and professional growth, representing a significant step towards success. The case method, a technique that lays the foundation for this content, ensures that the most current economic, social and professional reality is taken into account.

> *Our program prepares you to face new challenges in uncertain environments and achieve success in your career"*

The case method is the most widely used learning system in the best faculties in the world. The case method was developed in 1912 so that law students would not only learn the law based on theoretical content. It consisted of presenting students with real-life, complex situations for them to make informed decisions and value judgments on how to resolve them. In 1924, Harvard adopted it as a standard teaching method.

What should a professional do in a given situation? This is the question that you are presented with in the case method, an action-oriented learning method. Throughout the program, the studies will be presented with multiple real cases. They will have to combine all their knowledge and research, and argue and defend their ideas and decisions.

Relearning Methodology

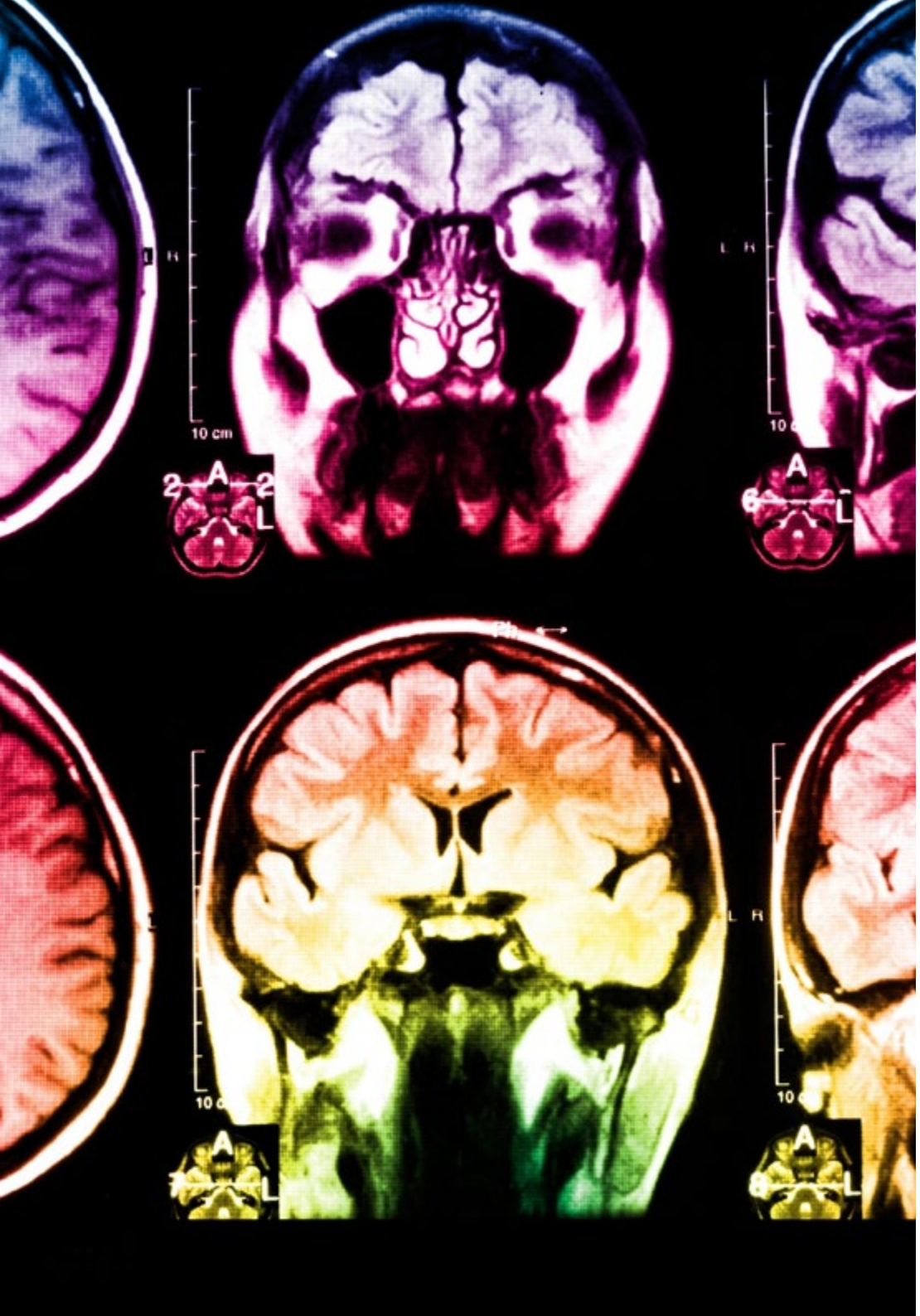
TECH effectively combines the Case Study methodology with a 100% online learning system based on repetition, which combines 8 different teaching elements in each lesson.

We enhance the Case Study with the best 100% online teaching method: Relearning.

In 2019, we obtained the best learning results of all online universities in the world.

At TECH, you will learn using a cutting-edge methodology designed to train the executives of the future. This method, at the forefront of international teaching, is called Relearning.

Our university is the only one in the world authorized to employ this successful method. In 2019, we managed to improve our students' overall satisfaction levels (teaching quality, quality of materials, course structure, objectives...) based on the best online university indicators.



In our program, learning is not a linear process, but rather a spiral (learn, unlearn, forget, and re-learn). Therefore, we combine each of these elements concentrically. This methodology has trained more than 650,000 university graduates with unprecedented success in fields as diverse as biochemistry, genetics, surgery, international law, management skills, sports science, philosophy, law, engineering, journalism, history, and financial markets and instruments. All this in a highly demanding environment, where the students have a strong socio-economic profile and an average age of 43.5 years.

Relearning will allow you to learn with less effort and better performance, involving you more in your training, developing a critical mindset, defending arguments, and contrasting opinions: a direct equation for success.

From the latest scientific evidence in the field of neuroscience, not only do we know how to organize information, ideas, images and memories, but we know that the place and context where we have learned something is fundamental for us to be able to remember it and store it in the hippocampus, to retain it in our long-term memory.

In this way, and in what is called neurocognitive context-dependent e-learning, the different elements in our program are connected to the context where the individual carries out their professional activity.

This program offers the best educational material, prepared with professionals in mind:

Study Material

All teaching material is produced by the specialists who teach the course, specifically for the course, so that the teaching content is highly specific and precise.

These contents are then applied to the audiovisual format, to create the TECH online working method. All this, with the latest techniques that offer high quality pieces in each and every one of the materials that are made available to the student.

Classes

There is scientific evidence suggesting that observing third-party experts can be useful.

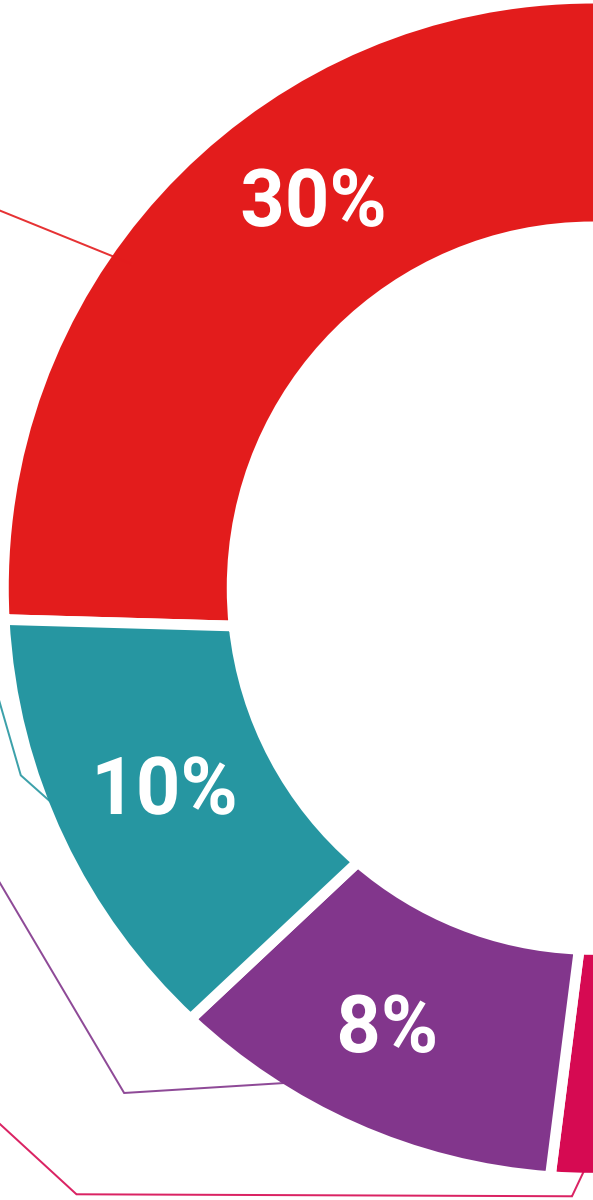
Learning from an Expert strengthens knowledge and memory, and generates confidence in future difficult decisions.

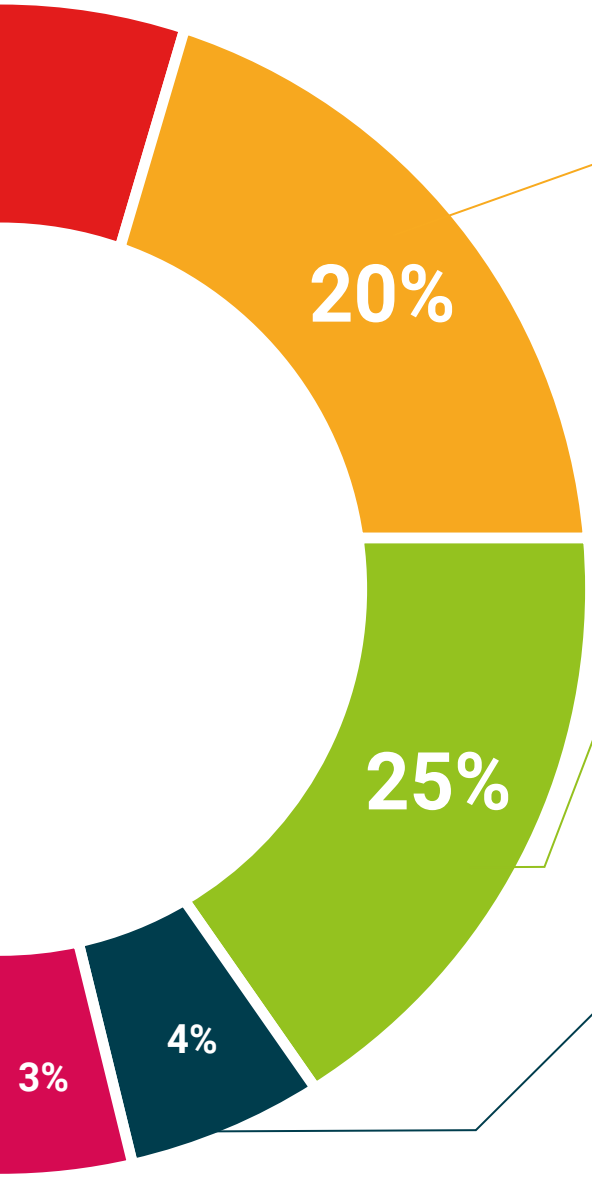
Practising Skills and Abilities

They will carry out activities to develop specific skills and abilities in each subject area. Exercises and activities to acquire and develop the skills and abilities that a specialist needs to develop in the context of the globalization that we are experiencing.

Additional Reading

Recent articles, consensus documents and international guidelines, among others. In TECH's virtual library, students will have access to everything they need to complete their course.





Case Studies

Students will complete a selection of the best case studies chosen specifically for this program. Cases that are presented, analyzed, and supervised by the best specialists in the world.

Interactive Summaries

The TECH team presents the contents attractively and dynamically in multimedia lessons that include audio, videos, images, diagrams, and concept maps in order to reinforce knowledge.

This exclusive educational system for presenting multimedia content was awarded by Microsoft as a "European Success Story".

Testing & Retesting

We periodically evaluate and re-evaluate students' knowledge throughout the program, through assessment and self-assessment activities and exercises, so that they can see how they are achieving their goals.

06

Certificate

The Professional Master's Degree in Artificial Intelligence and Knowledge Engineering guarantees students, in guarantees addition to the most rigorous and up-to-date education, access to a Professional Master's Degree issued by TECH Technological University.

Successfully complete this program and receive your university qualification without having to travel or fill out laborious paperwork"

This **Professional Master's Degree in Artificial Intelligence and Knowledge Engineering** contains the most complete and up-to-date program on the market.

After the student has passed the assessments, they will receive their corresponding **Professional Master's Degree** diploma issued by **TECH Technological University** via tracked delivery*.

The certificate issued by **TECH Technological University** will express the qualification obtained in the Professional Master's Degree, and meets the requirements commonly demanded by labor exchanges, competitive examinations, and professional career evaluation committees.

Title: **Professional Master's Degree in Artificial Intelligence and Knowledge Engineering**

Official Nº of Hours: **1,500 h.**


tech technological university

Awards the following

CERTIFICATE

to

Mr./Ms. _____, with identification number _____.

For having successfully passed and accredited the following program

PROFESSIONAL MASTER'S DEGREE

in

Artificial Intelligence and Knowledge Engineering

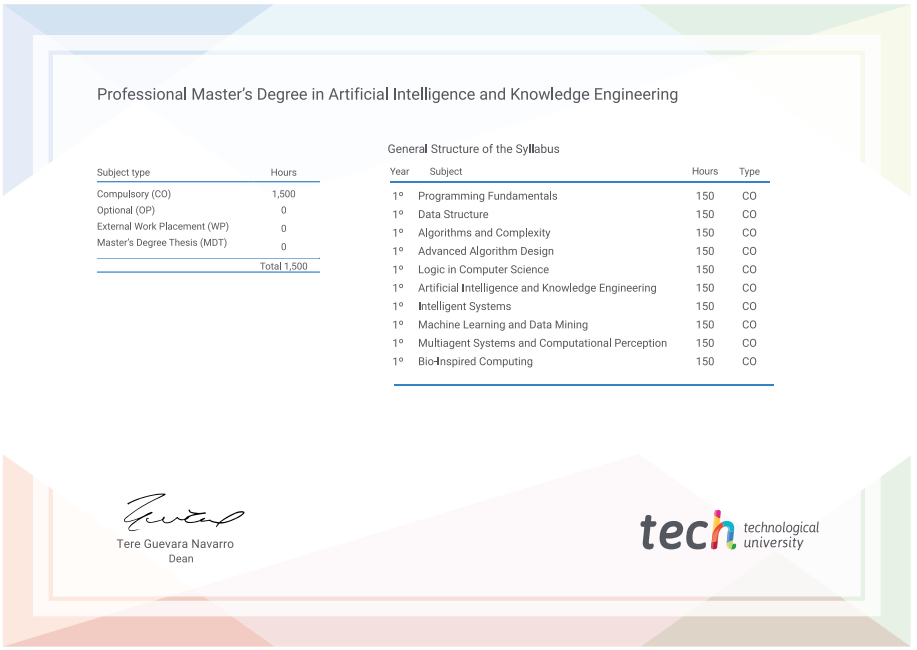
This is a qualification awarded by this University, equivalent to 1,500 hours, with a start date of dd/mm/yyyy and an end date of dd/mm/yyyy.

TECH is a Private Institution of Higher Education recognized by the Ministry of Public Education as of June 28, 2018.

June 17, 2020

Tere Guevara Navarro
Dean

This qualification must always be accompanied by the university degree issued by the competent authority to practice professionally in each country. Unique TECH Code: AFWORD23S techtitute.com/certificates


Professional Master's Degree in Artificial Intelligence and Knowledge Engineering

Subject type	Hours
Compulsory (CO)	1,500
Optional (OP)	0
External Work Placement (WP)	0
Master's Degree Thesis (MDT)	0
	Total 1,500

General Structure of the Syllabus

Year	Subject	Hours	Type
1º	Programming Fundamentals	150	CO
1º	Data Structure	150	CO
1º	Algorithms and Complexity	150	CO
1º	Advanced Algorithm Design	150	CO
1º	Logic in Computer Science	150	CO
1º	Artificial Intelligence and Knowledge Engineering	150	CO
1º	Intelligent Systems	150	CO
1º	Machine Learning and Data Mining	150	CO
1º	Multiagent Systems and Computational Perception	150	CO
1º	Bio-Inspired Computing	150	CO

Tere Guevara Navarro
Dean

tech technological university

*Apostille Convention. In the event that the student wishes to have their paper certificate issued with an apostille, TECH EDUCATION will make the necessary arrangements to obtain it, at an additional cost.

tech technological university

Professional Master's Degree

Artificial Intelligence and Knowledge Engineering

- Modality: **online**
- Duration: **12 months**
- Certificate: **TECH Technological University**
- Dedication: **16h/week**
- Schedule: **at your own pace**
- Exams: **online**

Professional Master's Degree

Artificial Intelligence and Knowledge Engineering

tech technological university