

# Master Privato

Intelligenza Artificiale e  
Ingegneria della Conoscenza





## Master Privato Intelligenza Artificiale e Ingegneria della Conoscenza

- » Modalità: **online**
- » Durata: **12 mesi**
- » Titolo: **TECH Università Tecnologica**
- » Dedizione: **16 ore/settimana**
- » Orario: **a scelta**
- » Esami: **online**

Accesso al sito web: [www.techitute.com/it/ingegneria/master/master-intelligenza-artificiale-ingegneria-conoscenza](http://www.techitute.com/it/ingegneria/master/master-intelligenza-artificiale-ingegneria-conoscenza)

# Indice

01

Presentazione

---

*pag. 4*

02

Obiettivi

---

*pag. 8*

03

Competenze

---

*pag. 14*

04

Struttura e contenuti

---

*pag. 18*

05

Metodologia

---

*pag. 30*

06

Titolo

---

*pag. 38*

# 01

# Presentazione

Il programma di Intelligenza Artificiale e Ingegneria della Conoscenza è progettato per far entrare i professionisti dell'ingegneria nell'entusiasmante mondo degli sviluppi in questa materia. Grazie a una preparazione altamente competitiva, potrai avanzare in modo solido e concreto in questo campo, acquisendo le competenze personali e professionali di un esperto necessarie per lavorare in questo settore. Un programma completo ed efficace che aumenterà la tua competenza fino ai massimi livelli.



“

*Diventa uno dei professionisti più richiesti del momento. Iscriviti al Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza e specializzati grazie a un esaustivo programma online"*

Gli sviluppi basati sull'intelligenza artificiale presentano numerose applicazioni nel campo dell'ingegneria. Dall'automazione di numerosi processi industriali e commerciali al controllo dei processi stessi. Ciò significa che i professionisti dell'ingegneria devono conoscere e padroneggiare il funzionamento di queste tecniche complesse.

Questa conoscenza essenziale è anche il primo passo per sviluppare le proprie capacità nel campo di questo tipo di tecnologia.

Nel corso della specializzazione viene offerto un panorama sulla realtà professionale al fine di poter valutare la convenienza della sua applicazione nel proprio progetto, valutandone le reali indicazioni, le modalità di sviluppo e le aspettative che si possono avere rispetto ai risultati.

Grazie all'esperienza, imparerai a sviluppare le competenze necessarie per progredire in questo settore di lavoro. Questo tipo di apprendimento, che richiede necessariamente esperienza, si concilia con l'e-learning e l'insegnamento pratico, offrendo un'opzione unica per dare al tuo CV la spinta che cercavi.



*Unisciti all'élite grazie a questa qualifica altamente efficace e accedi a nuovi percorsi per il tuo avanzamento professionale"*

Questo **Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza** possiede il programma più completo e aggiornato del mercato. Le caratteristiche principali del programma sono:

- ◆ Ultima tecnologia nel software di e-learning
- ◆ Sistema di insegnamento intensamente visivo, supportato da contenuti grafici e schematici di facile assimilazione e comprensione
- ◆ Sviluppo di casi di studio presentati da esperti attivi
- ◆ Sistemi di video interattivi di ultima generazione
- ◆ Insegnamento supportato dalla pratica online
- ◆ Sistemi di aggiornamento permanente
- ◆ Apprendimento autoregolato: piena compatibilità con altre occupazioni
- ◆ Esercizi pratici per l'autovalutazione e la verifica dell'apprendimento
- ◆ Gruppi di sostegno e sinergie educative: domande all'esperto, forum di discussione e conoscenza
- ◆ Comunicazione con l'insegnante e lavoro di riflessione individuale
- ◆ Contenuti disponibili da qualsiasi dispositivo fisso o mobile dotato di connessione a internet
- ◆ Banche di documentazione di supporto sempre disponibili, anche dopo il programma

“

*Un Master Privato che ti permetterà di lavorare in tutti i settori dell'Intelligenza Artificiale e dell'Ingegneria della Conoscenza con la capacità di un professionista di alto livello”*

Il nostro personale docente è composto da professionisti in diversi settori relazionati con questa specialità. In questo modo, offre la possibilità di raggiungere l'obiettivo di aggiornamento educativo a cui aspiriamo. Una squadra multidisciplinare di professionisti preparati ed esperti in diversi ambienti, che svilupperanno efficacemente le conoscenze teoriche ma, soprattutto, metteranno al tuo servizio le conoscenze pratiche derivate dalla propria esperienza: una delle qualità differenziali di questa specializzazione.

Questa padronanza della materia è completata dall'efficacia del disegno metodologico. Sviluppato da un team multidisciplinare di esperti di e-learning, integra gli ultimi progressi nella tecnologia educativa. In questo modo, potrai studiare con una serie di strumenti multimediali comodi e versatili che ti daranno l'operatività di cui hai bisogno per la tua specializzazione.

La creazione di questo programma è incentrata sull'Apprendimento Basato su Problemi, mediante il quale lo specialista deve cercare di risolvere le diverse situazioni che gli si presentano durante il corso. Al fine di raggiungere questo obiettivo in modalità remota viene usata la telepratica. Grazie all'aiuto di un innovativo sistema di video interattivi e del *Learning from an Expert*, potrai acquisire le conoscenze come se stessi affrontando i casi oggetto di studio in quel momento. Un concetto che permetterà di integrare e fissare l'apprendimento in modo più realistico e permanente.

*Unisciti all'élite grazie a questa qualifica di comprovata efficacia e accedi a nuovi percorsi per avanzare a livello professionale.*

*Grazie all'esperienza di professionisti attivi, esperti in Intelligenza Artificiale e Ingegneria della Conoscenza.*



02

# Obiettivi

Il nostro obiettivo è quello di preparare professionisti altamente qualificati per operare nel mondo del lavoro. Inoltre, promuoviamo in modo globale uno sviluppo umano che getti le basi per una società migliore. Questo obiettivo viene messo a portata di mano dei professionisti consentendogli di accedere a un livello maggiore di competenza e di controllo. Una meta che sarà possibile raggiungere in pochi mesi, grazie a una preparazione precisa e intensiva.



“

*Se il tuo obiettivo è quello di orientare le tue capacità verso nuovi percorsi di successo, questo è il programma che fa per te: una preparazione che punta all'eccellenza"*



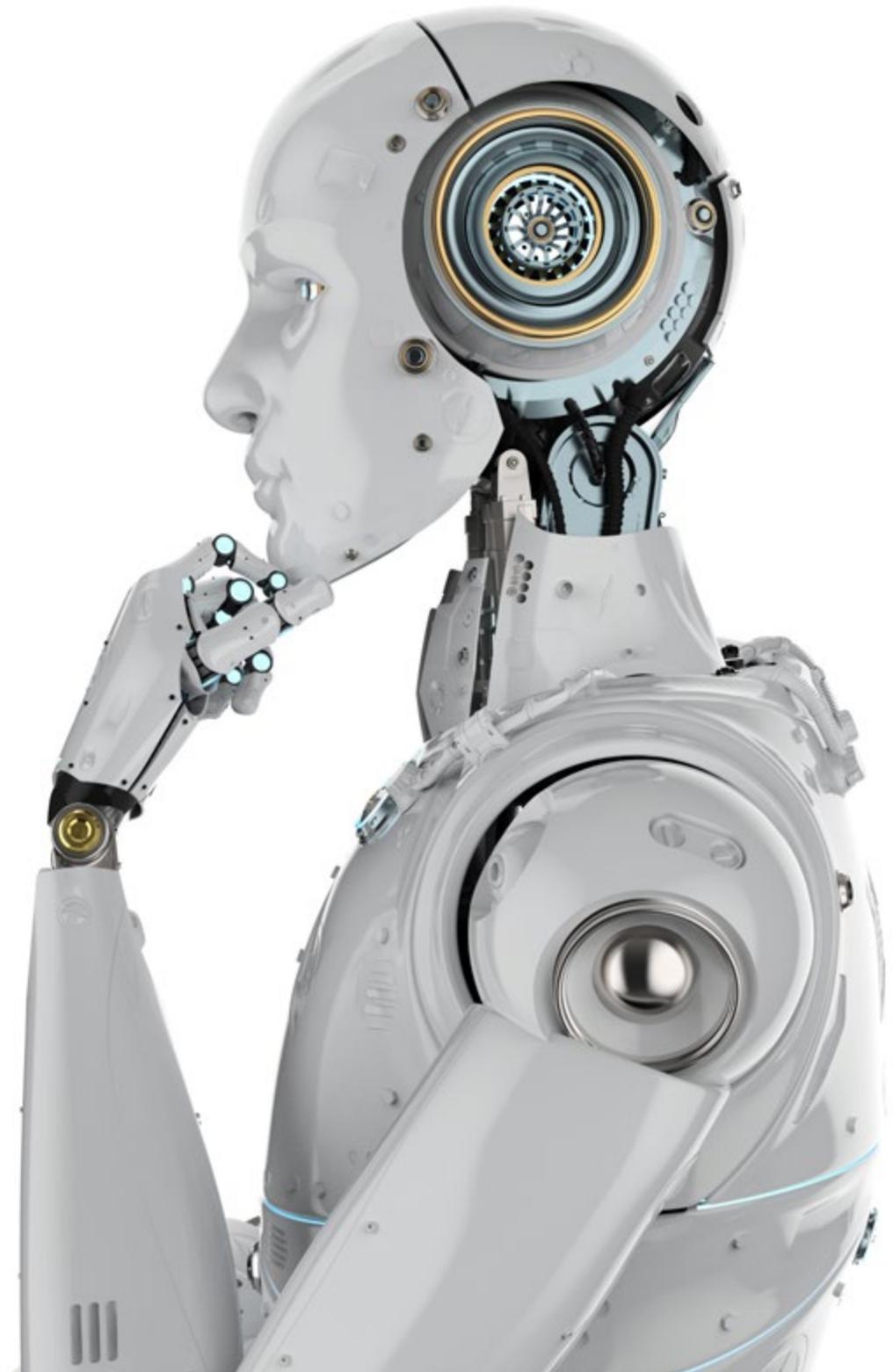
## Obiettivi generali

---

- ◆ Preparare scientificamente e tecnologicamente all'esercizio dell'Ingegneria informatica
- ◆ Ottenere una conoscenza completa nel campo dell'informatica
- ◆ Ottenere una conoscenza completa nel campo della struttura dei computer
- ◆ Acquisire le conoscenze necessarie per lavorare nell'ingegneria del software
- ◆ Rivedere le basi matematiche, statistiche e fisiche essenziali per questa materia

“

*Un percorso di specializzazione e crescita professionale che ti proietterà verso una maggiore competitività nel mercato del lavoro”*





## Obiettivi specifici

---

### Modulo 1. Fondamenti di programmazione

- ◆ Comprendere la struttura di base di un computer, il software e i linguaggi di programmazione di uso generale
- ◆ Imparare a progettare e interpretare gli algoritmi, che sono la base necessaria per lo sviluppo del software
- ◆ Comprendere gli elementi essenziali di un programma informatico, come i diversi tipi di dati, gli operatori, le espressioni, le dichiarazioni, le istruzioni di I/O e di controllo

### Modulo 2. Struttura dati

- ◆ Comprendere le diverse strutture di dati disponibili nei linguaggi di programmazione generici, sia statici che dinamici, e acquisire competenze essenziali nella gestione dei file
- ◆ Comprendere i tipi di dati astratti, i tipi di strutture dati lineari, le strutture dati gerarchiche semplici e complesse e la loro implementazione in C++
- ◆ Comprendere il funzionamento di strutture dati avanzate diverse da quelle abituali
- ◆ Comprendere la teoria e la pratica relative all'uso di heap e code prioritarie
- ◆ Imparare come funzionano le tabelle hash come tipi di dati astratti e funzioni

### Modulo 3. Algoritmi e complessità

- ◆ Comprendere le diverse tecniche di test del software e l'importanza di generare una buona documentazione insieme a un buon codice sorgente
- ◆ Imparare le basi del linguaggio di programmazione C++, uno dei più utilizzati al mondo
- ◆ Imparare le basi della programmazione in linguaggio C++, comprese classi, variabili, espressioni condizionali e oggetti
- ◆ Comprendere la teoria dei grafi e dei loro concetti avanzati, nonché degli algoritmi

### Modulo 4. Progettazione avanzata di algoritmi

- ◆ Imparare le principali strategie per la progettazione di algoritmi, nonché i diversi metodi e misure per il loro calcolo
- ◆ Conoscere i principali algoritmi di ordinamento utilizzati nello sviluppo del software
- ◆ Capire come funzionano i diversi algoritmi su alberi, heap e grafi
- ◆ Comprendere il funzionamento degli algoritmi Greedy, la loro strategia e gli esempi del loro utilizzo nei principali problemi noti. Conoscere anche l'uso degli algoritmi Greedy sui grafi
- ◆ Imparare le principali strategie di ricerca dei percorsi minimi, con l'approccio ai problemi essenziali del campo e agli algoritmi per la loro risoluzione
- ◆ Comprendere la tecnica del backtracking e i suoi principali utilizzi, nonché le tecniche alternative
- ◆ Approfondire la progettazione di algoritmi avanzati, analizzando algoritmi ricorsivi e tipo divide et impera, nonché eseguendo analisi ammortizzate
- ◆ Comprendere i concetti di programmazione dinamica e gli algoritmi per i problemi NP
- ◆ Comprendere il funzionamento dell'ottimizzazione combinatoria, i diversi algoritmi randomizzati e gli algoritmi paralleli
- ◆ Conoscere e capire come funzionano i diversi metodi di ricerca locale e con candidati

### Modulo 5. Logica computazionale

- ♦ Imparare i meccanismi della verifica formale dei programmi e di programmi iterativi, compresa la logica del primo ordine e il sistema formale di Hoare
- ♦ Imparare il funzionamento di alcuni dei principali metodi numerici come il metodo di bisezione, il metodo di Newton Raphson e il metodo della secante
- ♦ Imparare le basi della logica computazionale, il suo utilizzo e la sua giustificazione
- ♦ Conoscere le diverse strategie di formalizzazione e deduzione della logica proposizionale, tra cui il ragionamento naturale, la deduzione assiomatica e naturale, nonché le regole primitive del calcolo proposizionale
- ♦ Acquisire una conoscenza avanzata della logica proposizionale, approfondendo la sua semantica e le sue principali applicazioni, come i circuiti logici
- ♦ Comprendere la logica dei predicati sia per il loro calcolo di deduzione naturale sia per le strategie di formalizzazione e deduzione
- ♦ Comprendere le basi del linguaggio naturale e il suo meccanismo deduttivo

### Modulo 6. Intelligenza Artificiale e Ingegneria della Conoscenza

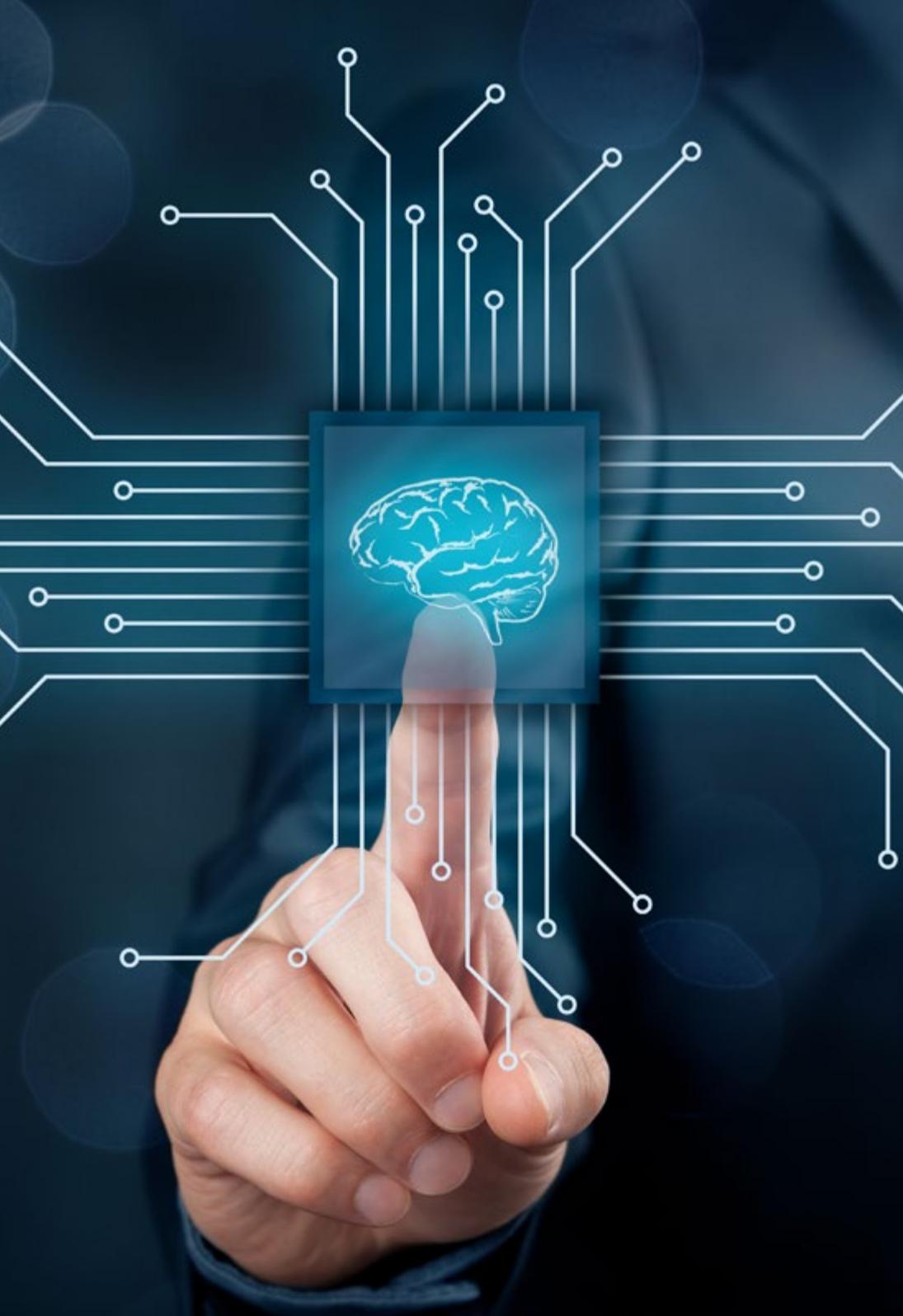
- ♦ Gettare le basi dell'intelligenza artificiale e dell'ingegneria della conoscenza, dando una breve occhiata alla storia dell'intelligenza artificiale fino ai giorni nostri
- ♦ Comprendere i concetti essenziali della ricerca nell'intelligenza artificiale, sia della ricerca informata che di quella non informata
- ♦ Capire come funziona l'intelligenza artificiale nei giochi
- ♦ Imparare i concetti fondamentali delle reti neurali e l'uso degli algoritmi genetici
- ♦ Acquisire i meccanismi appropriati per rappresentare la conoscenza, soprattutto tenendo conto del web semantico
- ♦ Comprendere il funzionamento dei sistemi esperti e dei sistemi di supporto alle decisioni

### Modulo 7. Sistemi intelligenti

- ♦ Imparare i concetti relativi alla teoria e all'architettura degli agenti e al loro processo di ragionamento
- ♦ Assimilare la teoria e la pratica dei concetti di informazione e conoscenza, nonché i diversi modi di rappresentare la conoscenza
- ♦ Comprendere la teoria relativa alle ontologie e imparare i linguaggi ontologici e i software per la creazione di ontologie
- ♦ Apprendere diversi modelli di rappresentazione della conoscenza, come vocabolari, tassonomie, thesauri e mappe mentali, tra gli altri
- ♦ Comprendere il funzionamento dei ragionatori semantici, dei sistemi basati sulla conoscenza e dei sistemi esperti
- ♦ Conoscere il funzionamento del web semantico, il suo stato attuale e futuro, nonché le applicazioni basate su di esso

### Modulo 8. Apprendimento automatico e data mining

- ♦ Favorire un approccio ai processi di scoperta della conoscenza e ai concetti di base dell'apprendimento automatico
- ♦ Imparare i metodi di esplorazione e pre-elaborazione dei dati e i diversi algoritmi basati sugli alberi decisionali
- ♦ Comprendere il funzionamento dei metodi bayesiani, di regressione e di risposta continua
- ♦ Comprendere le diverse regole di classificazione e la valutazione dei classificatori imparando a utilizzare le matrici di confusione e la valutazione numerica, la statistica Kappa e la curva ROC
- ♦ Acquisire le conoscenze essenziali relative al text mining e all'elaborazione del linguaggio naturale (NLP) e al clustering
- ♦ Approfondire la conoscenza delle reti neurali, dalle semplici fino alle ricorrenti



### **Modulo 9. Sistemi multiagente e percezione computazionale**

- ◆ Comprendere i concetti di base e avanzati relativi agli agenti e ai sistemi multi-agente
- ◆ Studiare lo standard degli agenti FIPA, prendendo in considerazione, tra le altre cose, la comunicazione tra di essi, la loro gestione e architettura
- ◆ Approfondire l'apprendimento della piattaforma JADE (Java Agent DEvelopment Framework), imparando a programmare in essa sia i concetti di base che quelli avanzati, compresi i temi della comunicazione e della scoperta degli agenti
- ◆ Gettare le basi dell'elaborazione del linguaggio naturale, come il riconoscimento automatico del parlato e la linguistica computazionale
- ◆ Conoscere in maniera approfondita la visione artificiale, l'analisi delle immagini digitali, a la loro trasformazione e segmentazione

### **Modulo 10. Informatica bio-ispirata**

- ◆ Introdurre il concetto di informatica bio-ispirata e comprendere il funzionamento di diversi tipi di algoritmi di adattamento sociale e di algoritmi genetici
- ◆ Approfondire lo studio dei diversi modelli di calcolo evolutivo, conoscendone le strategie, la programmazione, gli algoritmi e i modelli basati sulla stima delle distribuzioni
- ◆ Comprendere le principali strategie di esplorazione e sfruttamento dello spazio per gli algoritmi genetici
- ◆ Comprendere il funzionamento della programmazione evolutiva applicata a problemi di apprendimento e a problemi multi-obiettivo
- ◆ Imparare i concetti essenziali relativi alle reti neurali e capire come funzionano in casi d'uso reali applicati ad aree diverse come la ricerca medica, l'economia e la visione artificiale

# 03

# Competenze

Questo Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza è stato creato come strumento per fornire una qualifica di prim'ordine al professionista. La preparazione intensiva offerta preparerà a lavorare in tutti i settori legati alla sicurezza con le abilità proprie di un esperto del campo.



“

*Il Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza ti fornirà le competenze personali e professionali imprescindibili per svolgere un adeguato ruolo in ogni situazione lavorativa di questo ambito di intervento”*



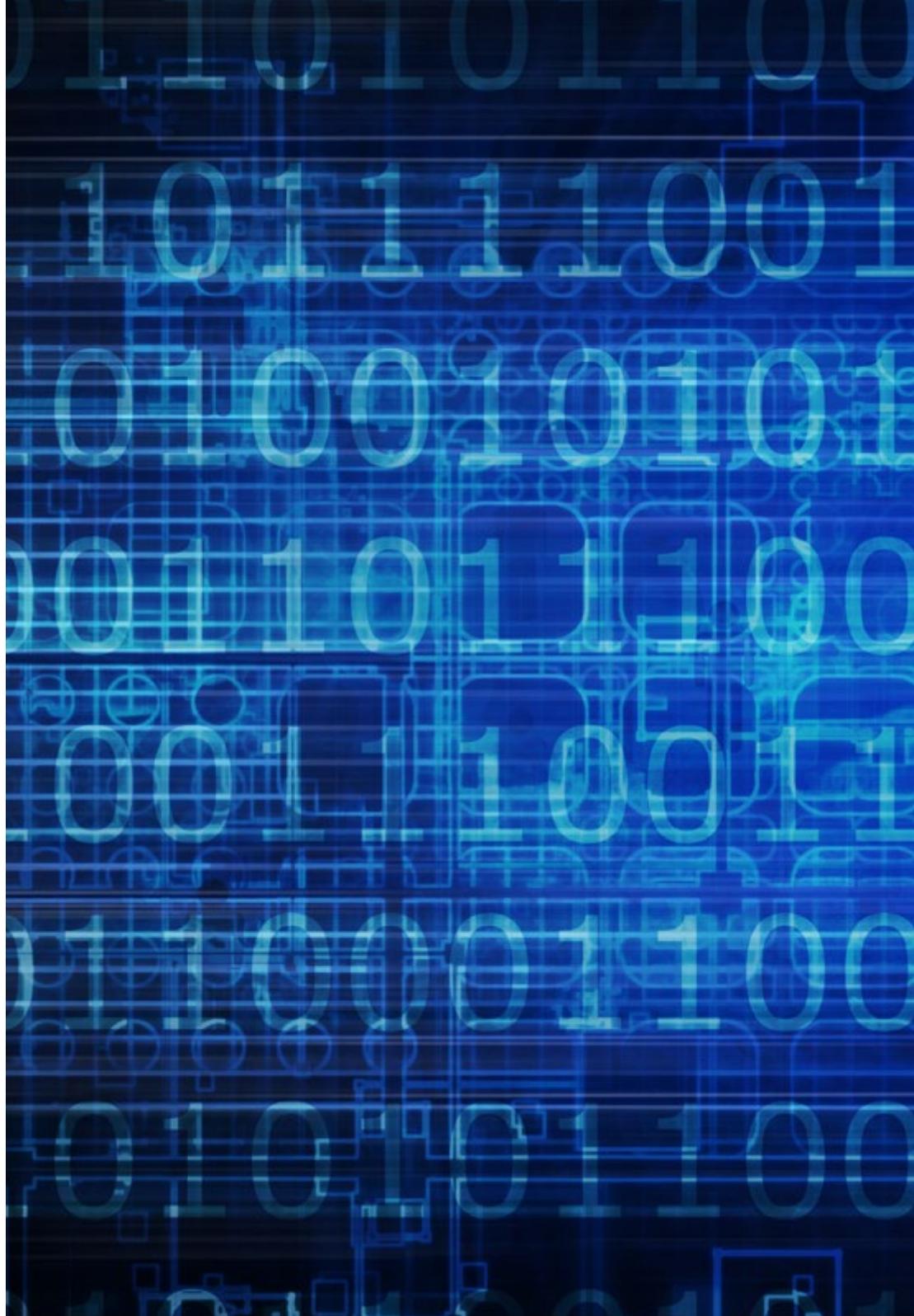
## Competenza generale

---

- ◆ Acquisire le competenze necessarie a un esercizio professionale nel campo dell'ingegneria informatica basate sulla conoscenza dei fattori essenziali per svolgerla con qualità ed efficienza

“

*Un'esperienza unica, chiave e decisiva per incrementare il tuo sviluppo professionale”*





## Competenze specifiche

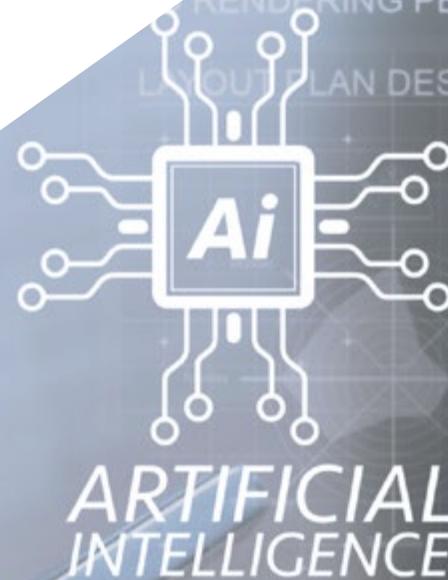
---

- ◆ Eseguire la programmazione nell'area dell'intelligenza artificiale tenendo conto di tutti i fattori di sviluppo dell'intelligenza artificiale
- ◆ Conoscere a fondo la struttura dei dati nella programmazione in C++
- ◆ Progettare algoritmi di base e avanzati
- ◆ Comprendere la logica computazionale e applicarla nella progettazione
- ◆ Conoscere l'intelligenza artificiale, i suoi usi e sviluppi e realizzare progetti propri
- ◆ Sapere cosa sono, come funzionano e come lavorare con i sistemi intelligenti
- ◆ Padroneggiare le basi dell'apprendimento automatico
- ◆ Saper usare JADE, FIPA, Machine Vision e altri sistemi multi-agente
- ◆ Comprendere gli algoritmi di calcolo bioispirati e le strategie per il loro utilizzo

# 04

## Struttura e contenuti

I contenuti di questo programma sono stati sviluppati da diversi specialisti con un chiaro obiettivo: permettere agli studenti di acquisire le abilità necessarie per diventare dei veri e propri esperti in materia. Un programma completo e ben strutturato che ti eleverà ai più alti standard di qualità e successo.



MA  
LEA



“

*Un programma d'insegnamento molto completo, strutturato in unità didattiche ben sviluppate, orientato a un apprendimento compatibile con il tuo stile di vita professionale”*

## Modulo 1. Fondamenti di programmazione

- 1.1. Introduzione alla programmazione
  - 1.1.1. Struttura base di un computer
  - 1.1.2. Software
  - 1.1.3. Linguaggi di programmazione
  - 1.1.4. Ciclo di vita di un'applicazione informatica
- 1.2. Progettazione di algoritmi
  - 1.2.1. Risoluzione dei problemi
  - 1.2.2. Tecniche descrittive
  - 1.2.3. Elementi e struttura di un algoritmo
- 1.3. Elementi di un programma
  - 1.3.1. Origine e caratteristiche del linguaggio C++
  - 1.3.2. Ambienti di sviluppo
  - 1.3.3. Concetto di programma
  - 1.3.4. Tipi di dati fondamentali
  - 1.3.5. Operatori
  - 1.3.6. Espressioni
  - 1.3.7. Strutture
  - 1.3.8. Ingresso e uscita di dati
- 1.4. Strutture di controllo
  - 1.4.1. Strutture
  - 1.4.2. Biforcazioni
  - 1.4.3. Loop
- 1.5. Astrazione e modularità: le funzioni
  - 1.5.1. Progettazione modulare
  - 1.5.2. Concetto di funzione e utilità
  - 1.5.3. Definizione di una funzione
  - 1.5.4. Flusso di esecuzione in una chiamata di funzione
  - 1.5.5. Prototipo di una funzione
  - 1.5.6. Restituzione dei risultati
  - 1.5.7. Chiamata di una funzione: parametri
  - 1.5.8. Passaggio di parametri per riferimento e per valore
  - 1.5.9. Ambito di identificazione



- 1.6. Strutture dati statiche
  - 1.6.1. *Array*
  - 1.6.2. Matrici. Poliedri
  - 1.6.3. Ricerca e ordinamento
  - 1.6.4. Stringhe. Funzioni di I/O per le catene
  - 1.6.5. Strutture. Unioni
  - 1.6.6. Nuovi tipi di dati
- 1.7. Strutture dati dinamiche: puntatori
  - 1.7.1. Concetto. Definizione di puntatore
  - 1.7.2. Operatori e operazioni con i puntatori
  - 1.7.3. *Array* di puntatori
  - 1.7.4. Puntatori e *Arrays*
  - 1.7.5. Puntatori a stringhe
  - 1.7.6. Puntatori a strutture
  - 1.7.7. Indirizzione multipla
  - 1.7.8. Puntatori a funzioni
  - 1.7.9. Passaggio di funzioni, strutture e *Array* come parametri di funzione
- 1.8. Cartelle
  - 1.8.1. Concetti di base
  - 1.8.2. Operazioni sui file
  - 1.8.3. Tipi di cartelle
  - 1.8.4. Organizzazione dei file
  - 1.8.5. Introduzione ai file C++
  - 1.8.6. Gestione dei file
- 1.9. Ricorsione
  - 1.9.1. Definizione di ricorsione
  - 1.9.2. Tipi di ricorsione
  - 1.9.3. Vantaggi e svantaggi
  - 1.9.4. Considerazioni
  - 1.9.5. Conversione ricorsiva-iterativa
  - 1.9.6. Lo stack di ricorsione

- 1.10. Prove e documentazione
  - 1.10.1. Test del programma
  - 1.10.2. Test della scatola bianca
  - 1.10.3. Test della scatola nera
  - 1.10.4. Strumenti per i test
  - 1.10.5. Documentazione del programma

## Modulo 2. Struttura dei dati

- 2.1. Introduzione alla programmazione C++
  - 2.1.1. Classi, costruttori, metodi e attributi
  - 2.1.2. Variabili
  - 2.1.3. Espressioni condizionali e loop
  - 2.1.4. Obiettivi
- 2.2. Tipi di dati astratti (ADT)
  - 2.2.1. Tipi di dati
  - 2.2.2. Strutture di base e ADT
  - 2.2.3. Vettori e *array*
- 2.3. Strutture dati lineari
  - 2.3.1. ADT Lista. Definizione
  - 2.3.2. Liste collegate e doppiamente collegate
  - 2.3.3. Liste ordinate
  - 2.3.4. Liste in C++
  - 2.3.5. ADT pila
  - 2.3.6. ADT coda
  - 2.3.7. Pila e coda in C++
- 2.4. Strutture dati gerarchiche
  - 2.4.1. ADT albero
  - 2.4.2. Percorsi
  - 2.4.3. Alberi n-ari
  - 2.4.4. Alberi binari
  - 2.4.5. Alberi binari di ricerca

- 2.5. Strutture dati gerarchiche: alberi complessi
  - 2.5.1. Alberi perfettamente bilanciati o di altezza minima
  - 2.5.2. Alberi multipercorso
  - 2.5.3. Riferimenti bibliografici
- 2.6. Heap e code di priorità
  - 2.6.1. ADT heap
  - 2.6.2. ADT coda di priorità
- 2.7. Tabelle Hash
  - 2.7.1. ADT Tabella Hash
  - 2.7.2. Funzioni Hash
  - 2.7.3. Funzioni Hash nelle tabelle Hash
  - 2.7.4. Ridispersione
  - 2.7.5. Tabelle Hash aperte
- 2.8. Grafi
  - 2.8.1. ADT grafo
  - 2.8.2. Tipi di grafo
  - 2.8.3. Rappresentazione grafica e operazioni di base
  - 2.8.4. Progettazione di grafi
- 2.9. Algoritmi e concetti avanzati sui grafi
  - 2.9.1. Problemi sui grafi
  - 2.9.2. Algoritmi sui percorsi
  - 2.9.3. Algoritmi di ricerca o percorsi
  - 2.9.4. Altri algoritmi
- 2.10. Altre strutture di dati
  - 2.10.1. Insiemi
  - 2.10.2. Array paralleli
  - 2.10.3. Tabelle di simboli
  - 2.10.4. Trie
- 3.2. Efficienza e analisi degli algoritmi
  - 3.2.1. Misure di efficienza
  - 3.2.2. Misurare le dimensioni dell'ingresso
  - 3.2.3. Misurare il tempo di esecuzione
  - 3.2.4. Caso peggiore, caso migliore e caso medio
  - 3.2.5. Notazione asintotica
  - 3.2.6. Criteri di analisi matematica per algoritmi non ricorsivi
  - 3.2.7. Analisi matematica di algoritmi ricorsivi
  - 3.2.8. Analisi empirica degli algoritmi
- 3.3. Algoritmi di ordinamento
  - 3.3.1. Concetto di gestione
  - 3.3.2. Ordinamento per bolla
  - 3.3.3. Ordinamento per selezione
  - 3.3.4. Ordinamento per inserimento
  - 3.3.5. Ordinamento per fusione (merge\_sort)
  - 3.3.6. Ordinamento rapido (quick\_sort)
- 3.4. Algoritmi con alberi
  - 3.4.1. Concetto di albero
  - 3.4.2. Alberi binari
  - 3.4.3. Percorsi di alberi
  - 3.4.4. Rappresentare espressioni
  - 3.4.5. Alberi binari ordinati
  - 3.4.6. Alberi binari bilanciati
- 3.5. Algoritmi con *Heap*
  - 3.5.1. Gli *Heap*
  - 3.5.2. L'algoritmo *HeapSort*
  - 3.5.3. Code di priorità
- 3.6. Algoritmi con grafi
  - 3.6.1. Rappresentazione
  - 3.6.2. Percorso in larghezza
  - 3.6.3. Percorso in profondità
  - 3.6.4. Ordinamento topologico

### Modulo 3. Algoritmi e complessità

- 3.1. Introduzione alle strategie di progettazione degli algoritmi
  - 3.1.1. Ricorsione
  - 3.1.2. Divide et impera
  - 3.1.3. Altre strategie

- 3.7. Algoritmi *Greedy*
  - 3.7.1. La strategia *Greedy*
  - 3.7.2. Elementi della strategia *Greedy*
  - 3.7.3. Scambio di monete
  - 3.7.4. Problema del viaggiatore
  - 3.7.5. Problema dello zaino
- 3.8. Ricerca di percorsi minimi
  - 3.8.1. Il problema del percorso minimo
  - 3.8.2. Archi negativi e cicli
  - 3.8.3. Algoritmo di Dijkstra
- 3.9. Algoritmi *Greedy* sui grafi
  - 3.9.1. L'albero di copertura minimo
  - 3.9.2. Algoritmo di Prim
  - 3.9.3. Algoritmo di Kruskal
  - 3.9.4. Analisi della complessità
- 3.10. *Backtracking*
  - 3.10.1. Il *Backtracking*
  - 3.10.2. Tecniche alternative

## Modulo 4. Progettazione avanzata di algoritmi

- 4.1. Analisi di algoritmi ricorsivi e divide et impera
  - 4.1.1. Porre e risolvere equazioni di ricorrenza omogenee e non omogenee
  - 4.1.2. Panoramica della strategia divide et impera
- 4.2. Analisi ammortizzata
  - 4.2.1. Analisi aggregata
  - 4.2.2. Il metodo di contabilità
  - 4.2.3. Il metodo del potenziale
- 4.3. Programmazione dinamica e algoritmi per problemi NP
  - 4.3.1. Caratteristiche della programmazione dinamica
  - 4.3.2. Indietro nel tempo: *Backtracking*
  - 4.3.3. Ramificazione e potatura
- 4.4. Ottimizzazione combinatoria
  - 4.4.1. Rappresentazione dei problemi
  - 4.4.2. Ottimizzazione 1D

- 4.5. Algoritmi randomizzati
  - 4.5.1. Esempi di algoritmi randomizzati
  - 4.5.2. Il teorema Buffon
  - 4.5.3. Algoritmo Monte Carlo
  - 4.5.4. Algoritmo Las Vegas
- 4.6. Ricerca locale e con candidati
  - 4.6.1. *Garcient Ascent*
  - 4.6.2. *Hill Climbing*
  - 4.6.3. *Simulated Annealing*
  - 4.6.4. *Tabu search*
  - 4.6.5. Ricerca con candidati
- 4.7. Verifica formale dei programmi
  - 4.7.1. Specifica delle astrazioni funzionali
  - 4.7.2. Il linguaggio della logica di prim'ordine
  - 4.7.3. Sistema formale di Hoare
- 4.8. Verifica dei programmi iterativi
  - 4.8.1. Regole del sistema formale di Hoare
  - 4.8.2. Concetto invariante di iterazioni
- 4.9. Metodi numerici
  - 4.9.1. Il metodo della bisezione
  - 4.9.2. Metodo di Newton Raphson
  - 4.9.3. Il metodo delle secanti
- 4.10. Algoritmi paralleli
  - 4.10.1. Operazioni binarie parallele
  - 4.10.2. Operazioni parallele con grafi
  - 4.10.3. Parallelismo nel divide et impera
  - 4.10.4. Parallelismo nella programmazione dinamica

## Modulo 5. Logica computazionale

- 5.1. Giustificazione della logica
  - 5.1.1. Oggetto di studio della logica
  - 5.1.2. A cosa serve la logica?
  - 5.1.3. Componenti e tipi di ragionamento
  - 5.1.4. Componenti di un calcolo logico
  - 5.1.5. Semantica
  - 5.1.6. Giustificazione dell'esistenza di una logica
  - 5.1.7. Come verificare che una logica sia adeguata?
- 5.2. Calcolo della deduzione naturale delle dichiarazioni
  - 5.2.1. Linguaggio formale
  - 5.2.2. Meccanismo deduttivo
- 5.3. Strategie di formalizzazione e deduzione per la logica proposizionale
  - 5.3.1. Strategie di formalizzazione
  - 5.3.2. Ragionamento naturale
  - 5.3.3. Leggi e regole
  - 5.3.4. Deduzione assiomatica e deduzione naturale
  - 5.3.5. Il calcolo della deduzione naturale
  - 5.3.6. Regole primitive del calcolo proposizionale
- 5.4. Semantica della logica proposizionale
  - 5.4.1. Tabelle di verità
  - 5.4.2. Equivalenze
  - 5.4.3. Tautologie e contraddizioni
  - 5.4.4. Convalida di frasi proposizionali
  - 5.4.5. Convalida mediante tabelle di verità
  - 5.4.6. Convalida mediante alberi semantici
  - 5.4.7. Convalida per confutazione
- 5.5. Applicazioni della logica proposizionale: circuiti logici
  - 5.5.1. Le porte di base
  - 5.5.2. Circuiti
  - 5.5.3. Modelli matematici dei circuiti
  - 5.5.4. Minimizzazione
  - 5.5.5. La seconda forma canonica e la forma minima nel prodotto di somme
  - 5.5.6. Altre porte

- 5.6. Calcolo della deduzione naturale dei predicati
  - 5.6.1. Linguaggio formale
  - 5.6.2. Meccanismo deduttivo
- 5.7. Strategie di formalizzazione per la logica dei predicati
  - 5.7.1. Introduzione alla formalizzazione nella logica dei predicati
  - 5.7.2. Strategie di formalizzazione con quantificatori
- 5.8. Strategie di deduzione per la logica dei predicati
  - 5.8.1. Motivo dell'omissione
  - 5.8.2. Presentazione delle nuove regole
  - 5.8.3. La logica dei predicati come calcolo di deduzione naturale
- 5.9. Applicazioni della logica dei predicati: introduzione alla programmazione logica
  - 5.9.1. Presentazione informale
  - 5.9.2. Elementi di Prolog
  - 5.9.3. Rivalutazione e cut-off
- 5.10. Teoria degli insiemi, logica dei predicati e relativa semantica
  - 5.10.1. Teoria intuizionistica degli insiemi
  - 5.10.2. Introduzione alla semantica dei predicati

## Modulo 6. Intelligenza Artificiale e Ingegneria della Conoscenza

- 6.1. Introduzione all'intelligenza artificiale e all'ingegneria della conoscenza
  - 6.1.1. Breve storia dell'intelligenza artificiale
  - 6.1.2. L'intelligenza artificiale oggi
  - 6.1.3. Ingegneria della conoscenza
- 6.2. Ricerca
  - 6.2.1. Concetti di ricerca comuni
  - 6.2.2. Ricerca non informata
  - 6.2.3. Ricerca informata
- 6.3. Soddisfacibilità booleana, soddisfacibilità dei vincoli e pianificazione automatica
  - 6.3.1. Soddisfacibilità booleana
  - 6.3.2. Problemi di soddisfazione dei vincoli
  - 6.3.3. Pianificazione automatica e PDDL
  - 6.3.4. La pianificazione come ricerca euristica
  - 6.3.5. Pianificazione con il SAT

- 6.4. Intelligenza artificiale nei giochi
  - 6.4.1. Teoria dei giochi
  - 6.4.2. Minimax e potatura Alfa-Beta
  - 6.4.3. Simulazione: Monte Carlo
- 6.5. Apprendimento supervisionato e non
  - 6.5.1. Introduzione all'apprendimento automatico
  - 6.5.2. Classificazione
  - 6.5.3. Regressione
  - 6.5.4. Convalida dei risultati
  - 6.5.5. Aggruppamento (*Clustering*)
- 6.6. Reti neurali
  - 6.6.1. Fondamenti biologici
  - 6.6.2. Modello computazionale
  - 6.6.3. Reti neurali supervisionate e non supervisionate
  - 6.6.4. Percettrone semplice
  - 6.6.5. Percettrone multistrato
- 6.7. Algoritmi genetici
  - 6.7.1. Storia
  - 6.7.2. Basi biologiche
  - 6.7.3. Codifica dei problemi
  - 6.7.4. Generazione della popolazione iniziale
  - 6.7.5. Algoritmo principale e operatori genetici
  - 6.7.6. Valutazione degli individui: fitness
- 6.8. Thesauri, vocabolari, tassonomie
  - 6.8.1. Vocabolari
  - 6.8.2. Tassonomie
  - 6.8.3. Thesauri
  - 6.8.4. Ontologie
- 6.9. Rappresentazione della conoscenza: web semantico
  - 6.9.1. Web semantico
  - 6.9.2. Specifiche: RDF, RDFS e OWL
  - 6.9.3. Inferenza/ragionamento
  - 6.9.4. Linked Data

- 6.10. Sistemi esperti e DSS
  - 6.10.1. Sistemi esperti
  - 6.10.2. Sistemi di supporto decisionale

## Modulo 7. Sistemi intelligenti

- 7.1. Teoria degli agenti
  - 7.1.1. Storia del concetto
  - 7.1.2. Definizione di agente
  - 7.1.3. Attori in Intelligenza Artificiale
  - 7.1.4. Attori nell'Ingegneria del Software
- 7.2. Architetture di agenti
  - 7.2.1. Il processo del ragionamento di un agente
  - 7.2.2. Agenti reattivi
  - 7.2.3. Agenti deduttivi
  - 7.2.4. Agenti ibridi
  - 7.2.5. Confronto
- 7.3. Informazione e conoscenza
  - 7.3.1. Distinzione tra dati, informazioni e conoscenza
  - 7.3.2. Valutazione della qualità dei dati
  - 7.3.3. Metodi di acquisizione dei dati
  - 7.3.4. Metodi di acquisizione delle informazioni
  - 7.3.5. Metodi di acquisizione delle conoscenze
- 7.4. Rappresentazione della conoscenza
  - 7.4.1. L'importanza dell'appropriazione delle conoscenze
  - 7.4.2. Definizione della rappresentazione della conoscenza attraverso i loro ruoli
  - 7.4.3. Caratteristiche di una rappresentazione delle conoscenze
- 7.5. Ontologie
  - 7.5.1. Introduzione ai metadati
  - 7.5.2. Concetto filosofico di ontologia
  - 7.5.3. Concetto informatico di ontologia
  - 7.5.4. Ontologie di dominio e ontologie di livello superiore
  - 7.5.5. Come costruire un'ontologia

- 7.6. Linguaggi ontologici e software per la creazione di ontologie
  - 7.6.1. Tripla RDF, Turtle e N3
  - 7.6.2. RDF Schema
  - 7.6.3. OWL
  - 7.6.4. SPARQL
  - 7.6.5. Introduzione ai diversi strumenti per la creazione di ontologie
  - 7.6.6. Installazione e utilizzo di Protégé
- 7.7. Web semantica
  - 7.7.1. Stato attuale e futuro del web semantico
  - 7.7.2. Applicazioni del web semantico
- 7.8. Altri modelli di rappresentazione delle conoscenze
  - 7.8.1. Vocabolari
  - 7.8.2. Visione globale
  - 7.8.3. Tassonomie
  - 7.8.4. Thesauri
  - 7.8.5. Folksonome
  - 7.8.6. Confronto
  - 7.8.7. Mappe mentali
- 7.9. Valutazione e integrazione delle rappresentazioni della conoscenza
  - 7.9.1. Logica dell'ordine zero
  - 7.9.2. Logica di primo ordine
  - 7.9.3. Logica descrittiva
  - 7.9.4. Relazione tra i diversi tipi di logica
  - 7.9.5. Prolog: programmazione basata sulla logica del primo ordine
- 7.10. Ragonatori semantici, sistemi basati sulla conoscenza e sistemi esperti
  - 7.10.1. Concetto di ragionatore
  - 7.10.2. Applicazioni di un ragionatore
  - 7.10.3. Sistemi basati sulla conoscenza
  - 7.10.4. MYCIN, storia dei sistemi esperti
  - 7.10.5. Elementi e architettura dei sistemi esperti
  - 7.10.6. Creazione di sistemi esperti

## Modulo 8. Apprendimento automatico e data mining

- 8.1. Introduzione ai processi di scoperta della conoscenza e ai concetti di base dell'apprendimento automatico
  - 8.1.1. Concetti chiave del processo di scoperta della conoscenza
  - 8.1.2. Prospettiva storica del processo di scoperta della conoscenza
  - 8.1.3. Tappe del processo di scoperta della conoscenza
  - 8.1.4. Tecniche utilizzate nel processo di scoperta della conoscenza
  - 8.1.5. Caratteristiche dei buoni modelli di apprendimento automatico
  - 8.1.6. Tipi di informazioni sull'apprendimento automatico
  - 8.1.7. Concetti base di apprendimento
  - 8.1.8. Concetti base di apprendimento non supervisionato
- 8.2. Esplorazione e pre-elaborazione dei dati
  - 8.2.1. Trattamento dei dati
  - 8.2.2. Elaborazione dei dati nel flusso di analisi dei dati
  - 8.2.3. Tipi di dati
  - 8.2.4. Trasformazioni dei dati
  - 8.2.5. Visualizzazione ed esplorazione di variabili continue
  - 8.2.6. Visualizzazione ed esplorazione di variabili categoriche
  - 8.2.7. Misure di correlazione
  - 8.2.8. Rappresentazioni grafiche più comuni
  - 8.2.9. Introduzione all'analisi multivariata e alla riduzione della dimensionalità
- 8.3. Albero decisionale
  - 8.3.1. Algoritmo ID3
  - 8.3.2. Algoritmo C4.5
  - 8.3.3. Overtraining e potatura
  - 8.3.4. Analisi dei risultati
- 8.4. Valutazione dei classificatori
  - 8.4.1. Matrici di confusione
  - 8.4.2. Matrici di valutazione numerica
  - 8.4.3. Statistica Kappa
  - 8.4.4. La curva ROC

- 8.5. Regole di classificazione
  - 8.5.1. Misure di valutazione delle regole
  - 8.5.2. Introduzione alle rappresentazioni grafiche
  - 8.5.3. Algoritmo di sovrapposizione sequenziale
- 8.6. Reti neurali
  - 8.6.1. Concetti di base
  - 8.6.2. Reti neurali semplici
  - 8.6.3. Algoritmo di *backpropagation*
  - 8.6.4. Introduzione alle reti neurali ricorrenti
- 8.7. Metodi bayesiani
  - 8.7.1. Concetti base di probabilità
  - 8.7.2. Teorema di Bayes
  - 8.7.3. Naive Bayes
  - 8.7.4. Introduzione alle reti bayesiane
- 8.8. Modelli di regressione e di risposta continua
  - 8.8.1. Regressione lineare semplice
  - 8.8.2. Regressione lineare multipla
  - 8.8.3. Regressione logistica
  - 8.8.4. Alberi di regressione
  - 8.8.5. Introduzione alle macchine a vettori di supporto (SVM)
  - 8.8.6. Misure di bontà dell'adattamento
- 8.9. *Clustering*
  - 8.9.1. Concetti di base
  - 8.9.2. Clustering gerarchico
  - 8.9.3. Metodi probabilistici
  - 8.9.4. Algoritmo EM
  - 8.9.5. Metodo B-Cubed
  - 8.9.6. Metodi impliciti
- 8.10. Text mining ed elaborazione del linguaggio naturale (NLP)
  - 8.10.1. Concetti di base
  - 8.10.2. Creazione del corpus
  - 8.10.3. Analisi descrittiva
  - 8.10.4. Introduzione all'analisi di sentimenti

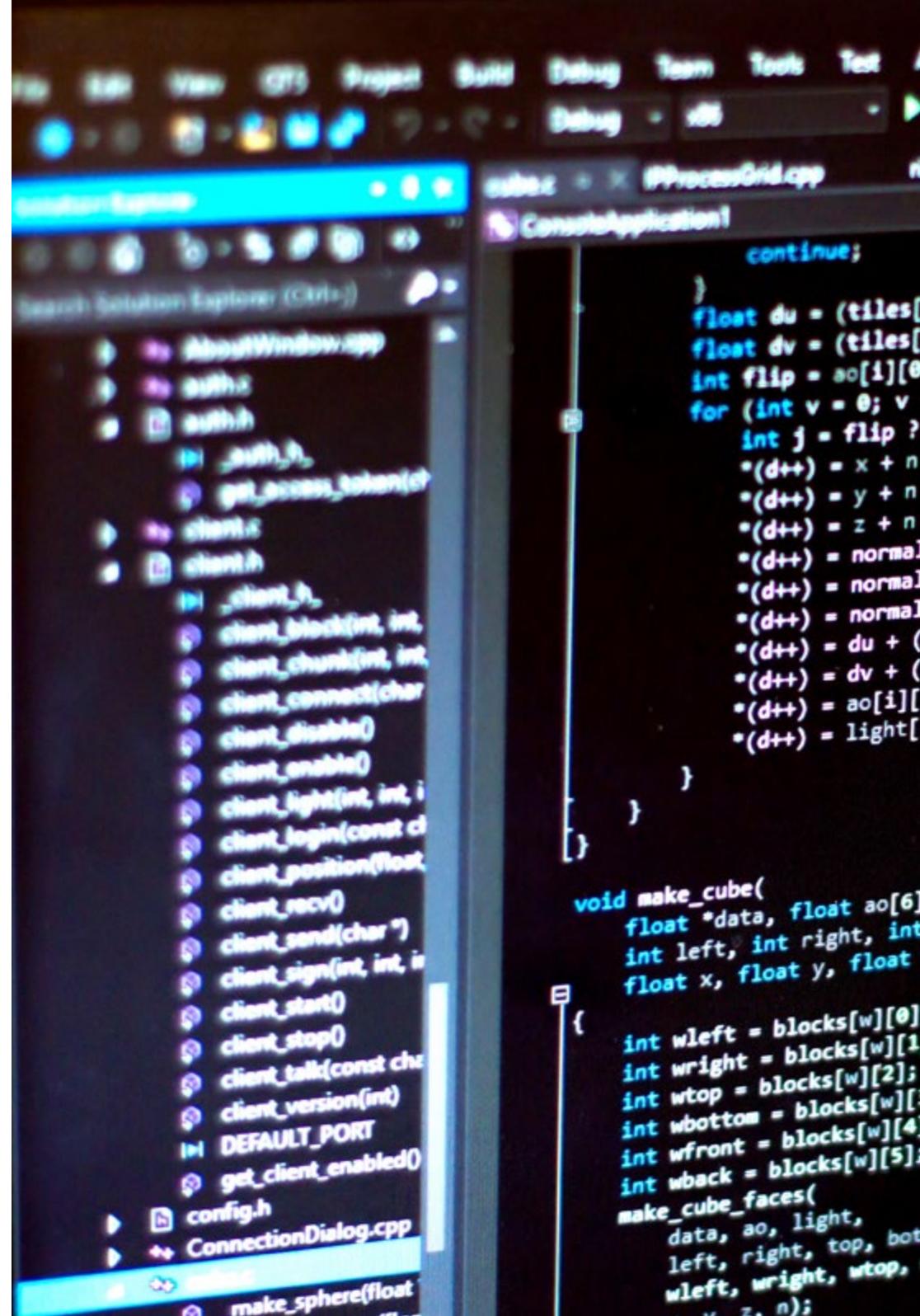
## Modulo 9. Sistemi multiagente e percezione computazionale

- 9.1. Agenti e sistemi multiagente
  - 9.1.1. Concetto di agente
  - 9.1.2. Architetture
  - 9.1.3. Comunicazione e coordinamento
  - 9.1.4. Linguaggi e strumenti di programmazione
  - 9.1.5. Applicazioni degli agenti
  - 9.1.6. La FIPA
- 9.2. Lo standard per gli agenti: FIPA
  - 9.2.1. Comunicazione tra gli agenti
  - 9.2.2. La gestione degli agenti
  - 9.2.3. Architettura astratta
  - 9.2.4. Altre specifiche
- 9.3. La piattaforma JADE
  - 9.3.1. Agenti software secondo JADE
  - 9.3.2. Architettura
  - 9.3.3. Installazione e implementazione
  - 9.3.4. Pacchetti JADE
- 9.4. Programmazione di base con JADE
  - 9.4.1. La console di gestione
  - 9.4.2. Creazione di agenti di base
- 9.5. Programmazione avanzata con JADE
  - 9.5.1. Creazione avanzata di agenti
  - 9.5.2. Comunicazione tra gli agenti
  - 9.5.3. Scoperta degli agenti
- 9.6. Visione Artificiale
  - 9.6.1. Elaborazione e analisi digitale delle immagini
  - 9.6.2. Analisi delle immagini e visione computerizzata
  - 9.6.3. Processo delle immagini e visione umana
  - 9.6.4. Sistemi di imaging
  - 9.6.5. Preparazione dell'immagine e percezione

- 9.7. Analisi di immagini digitali
  - 9.7.1. Fasi del processo di analisi delle immagini
  - 9.7.2. Preelaborazione
  - 9.7.3. Operazioni basilari
  - 9.7.4. Filtro spaziale
- 9.8. Trasformazione e segmentazione delle immagini digitali
  - 9.8.1. Trasformazione di Fourier
  - 9.8.2. Filtro a frequenze
  - 9.8.3. Concetti di base
  - 9.8.4. Soglia
  - 9.8.5. Rilevamento dei contorni
- 9.9. Riconoscimento della forma
  - 9.9.1. Estrazione delle caratteristiche
  - 9.9.2. Algoritmi di classificazione
- 9.10. Elaborazione di linguaggio naturale
  - 9.10.1. Riconoscimento vocale automatico
  - 9.10.2. Linguistica computazionale

## Modulo 10. Informatica bio-ispirata

- 10.1. Introduzione al Bio-Inspired Computing
  - 10.1.1. Introduzione al Bio-Inspired Computing
- 10.2. Algoritmi di adattamento sociale
  - 10.2.1. Informatica bio-ispirata basata su colonie di formiche
  - 10.2.2. Varianti degli algoritmi di colonia di formiche
  - 10.2.3. Computing basato su nubi di particelle
- 10.3. Algoritmi genetici
  - 10.3.1. Struttura generale
  - 10.3.2. Implementazioni dei principali operatori
- 10.4. Strategie di esplorazione e sfruttamento dello spazio per gli algoritmi genetici
  - 10.4.1. Algoritmo CHC
  - 10.4.2. Problemi multimodali



```
    i] % 16) * s;  
    i] / 16) * s;  
    ] + ao[i][3] > ao[i][1] + ao[i][2];  
    < 6; v++) {  
        flipped[i][v] : indices[i][v];  
        * positions[i][j][0];  
        * positions[i][j][1];  
        * positions[i][j][2];  
        ls[i][0];  
        ls[i][1];  
        ls[i][2];  
        uvs[i][j][0] ? b : a);  
        uvs[i][j][1] ? b : a);  
        j];  
        i][j];
```

```
[4], float light[6][4],  
: top, int bottom, int front, int back,  
z, float n, int w)
```

```
;  
];
```

```
3];  
];  
;
```

```
tom, front, back,  
ubottom, ufront, uback,
```

- 10.5. Modelli di computing evolutivo (I)
  - 10.5.1. Strategie evolutive
  - 10.5.2. Programmazione evolutiva
  - 10.5.3. Algoritmi basati sull'evoluzione differenziale
- 10.6. Modelli di computing evolutivo (II)
  - 10.6.1. Modelli di evoluzione basati sulla stima delle distribuzioni
  - 10.6.2. Programmazione genetica
- 10.7. Programmazione dello sviluppo applicata ai disturbi dell'apprendimento
  - 10.7.1. Apprendimento basato sulle regole
  - 10.7.2. Metodi evolutivi nei problemi di selezione delle istanze
- 10.8. Problemi multi-obiettivo
  - 10.8.1. Concetto di dominanza
  - 10.8.2. Applicazione di algoritmi evolutivi a problemi multi-obiettivo
- 10.9. Reti neurali (I)
  - 10.9.1. Introduzione alle reti neurali
  - 10.9.2. Esempio pratico con le reti neurali
- 10.10. Reti neurali (II)
  - 10.10.1. Casi di utilizzo delle reti neurali nella ricerca medica
  - 10.10.2. Casi di utilizzo delle reti neurali nell'economia
  - 10.10.3. Casi di utilizzo delle reti neurali nella visione artificiale

05

# Metodologia

Questo programma ti offre un modo differente di imparare. La nostra metodologia si sviluppa in una modalità di apprendimento ciclico: *il Relearning*.

Questo sistema di insegnamento viene applicato nelle più prestigiose facoltà di medicina del mondo ed è considerato uno dei più efficaci da importanti pubblicazioni come il *New England Journal of Medicine*.



“

*Scopri il Relearning, un sistema che abbandona l'apprendimento lineare convenzionale, per guidarti attraverso dei sistemi di insegnamento ciclici: una modalità di apprendimento che ha dimostrato la sua enorme efficacia, soprattutto nelle materie che richiedono la memorizzazione”*

## Caso di Studio per contestualizzare tutti i contenuti

Il nostro programma offre un metodo rivoluzionario per sviluppare le abilità e le conoscenze. Il nostro obiettivo è quello di rafforzare le competenze in un contesto mutevole, competitivo e altamente esigente.

“

*Con TECH potrai sperimentare un modo di imparare che sta scuotendo le fondamenta delle università tradizionali in tutto il mondo"*



*Avrai accesso a un sistema di apprendimento basato sulla ripetizione, con un insegnamento naturale e progressivo durante tutto il programma.*



*Imparerai, attraverso attività collaborative e casi reali, la risoluzione di situazioni complesse in ambienti aziendali reali.*

## Un metodo di apprendimento innovativo e differente

Questo programma di TECH consiste in un insegnamento intensivo, creato ex novo, che propone le sfide e le decisioni più impegnative in questo campo, sia a livello nazionale che internazionale. Grazie a questa metodologia, la crescita personale e professionale viene potenziata, effettuando un passo decisivo verso il successo. Il metodo casistico, la tecnica che sta alla base di questi contenuti, garantisce il rispetto della realtà economica, sociale e professionale più attuali.

“ *Il nostro programma ti prepara ad affrontare nuove sfide in ambienti incerti e a raggiungere il successo nella tua carriera* ”

Il metodo casistico è stato il sistema di apprendimento più usato nelle migliori facoltà del mondo. Sviluppato nel 1912 affinché gli studenti di Diritto non imparassero la legge solo sulla base del contenuto teorico, il metodo casistico consisteva nel presentare loro situazioni reali e complesse per prendere decisioni informate e giudizi di valore su come risolverle. Nel 1924 fu stabilito come metodo di insegnamento standard ad Harvard.

Cosa dovrebbe fare un professionista per affrontare una determinata situazione? Questa è la domanda con cui ti confrontiamo nel metodo dei casi, un metodo di apprendimento orientato all'azione. Durante il programma, gli studenti si confronteranno con diversi casi di vita reale. Dovranno integrare tutte le loro conoscenze, effettuare ricerche, argomentare e difendere le proprie idee e decisioni.

## Metodologia Relearning

TECH coniuga efficacemente la metodologia del Caso di Studio con un sistema di apprendimento 100% online basato sulla ripetizione, che combina 8 diversi elementi didattici in ogni lezione.

Potenziamo il Caso di Studio con il miglior metodo di insegnamento 100% online: il Relearning.

*Nel 2019 abbiamo ottenuto i migliori risultati di apprendimento di tutte le università online del mondo.*

In TECH si impara attraverso una metodologia all'avanguardia progettata per formare i manager del futuro. Questo metodo, all'avanguardia della pedagogia mondiale, si chiama Relearning.

La nostra università è l'unica autorizzata a utilizzare questo metodo di successo. Nel 2019, siamo riusciti a migliorare il livello di soddisfazione generale dei nostri studenti (qualità dell'insegnamento, qualità dei materiali, struttura del corso, obiettivi...) rispetto agli indicatori della migliore università online.





Nel nostro programma, l'apprendimento non è un processo lineare, ma avviene in una spirale (impariamo, disimpariamo, dimentichiamo e re-impariamo). Pertanto, combiniamo ciascuno di questi elementi in modo concentrico. Questa metodologia ha formato più di 650.000 laureati con un successo senza precedenti in campi diversi come la biochimica, la genetica, la chirurgia, il diritto internazionale, le competenze manageriali, le scienze sportive, la filosofia, il diritto, l'ingegneria, il giornalismo, la storia, i mercati e gli strumenti finanziari. Tutto questo in un ambiente molto esigente, con un corpo di studenti universitari con un alto profilo socio-economico e un'età media di 43,5 anni.

*Il Relearning ti permetterà di apprendere con meno sforzo e più performance, impegnandoti maggiormente nella tua specializzazione, sviluppando uno spirito critico, difendendo gli argomenti e contrastando le opinioni: un'equazione diretta al successo.*

Dalle ultime evidenze scientifiche nel campo delle neuroscienze, non solo sappiamo come organizzare le informazioni, le idee, le immagini e i ricordi, ma sappiamo che il luogo e il contesto in cui abbiamo imparato qualcosa è fondamentale per la nostra capacità di ricordarlo e immagazzinarlo nell'ippocampo, per conservarlo nella nostra memoria a lungo termine.

In questo modo, e in quello che si chiama Neurocognitive Context-dependent E-learning, i diversi elementi del nostro programma sono collegati al contesto in cui il partecipante sviluppa la sua pratica professionale.

Questo programma offre i migliori materiali didattici, preparati appositamente per i professionisti:



#### Materiali di studio

Tutti i contenuti didattici sono creati appositamente per il corso dagli specialisti che lo impartiranno, per fare in modo che lo sviluppo didattico sia davvero specifico e concreto.

Questi contenuti sono poi applicati al formato audiovisivo che supporterà la modalità di lavoro online di TECH. Tutto questo, con le ultime tecniche che offrono componenti di alta qualità in ognuno dei materiali che vengono messi a disposizione dello studente.



#### Master class

Esistono evidenze scientifiche sull'utilità dell'osservazione di esperti terzi.

Imparare da un esperto rafforza la conoscenza e la memoria, costruisce la fiducia nelle nostre future decisioni difficili.



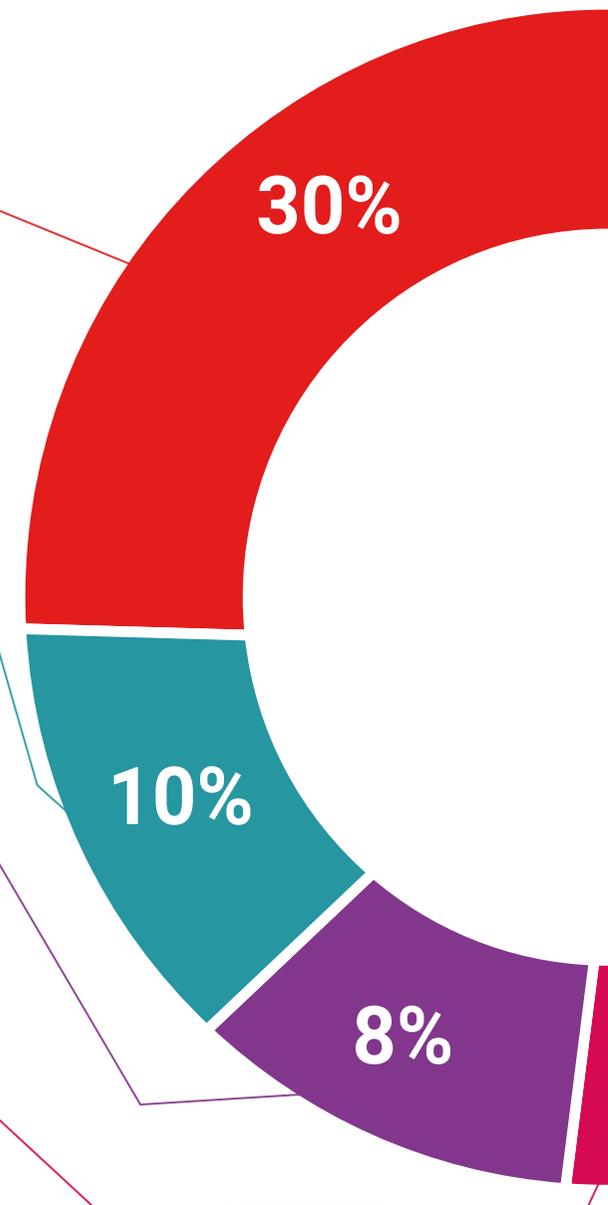
#### Pratiche di competenze e competenze

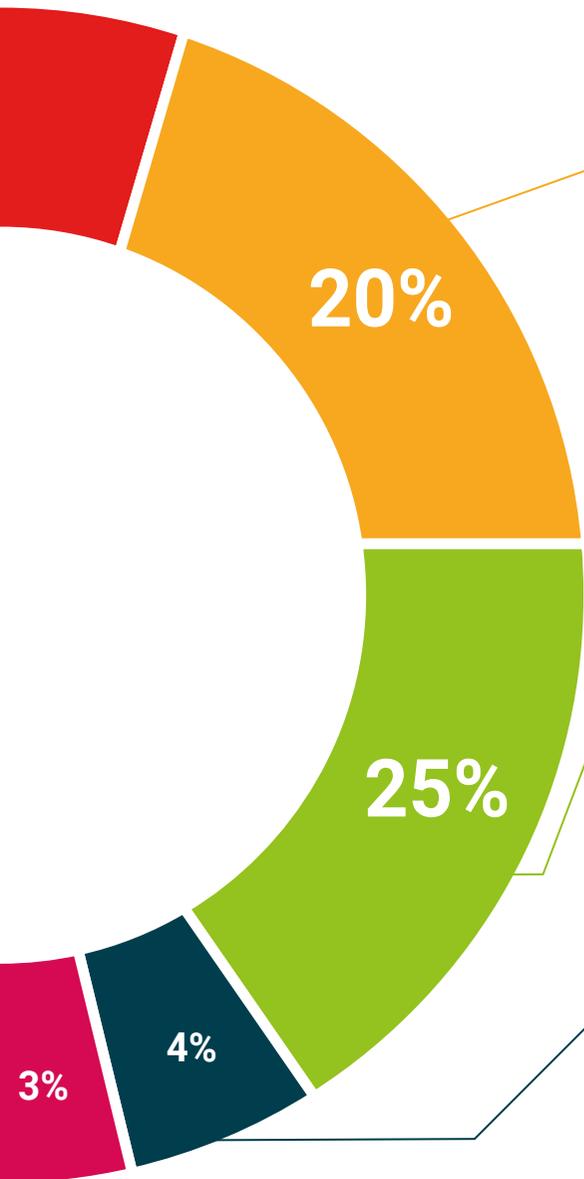
Svolgerai attività per sviluppare competenze e capacità specifiche in ogni area tematica. Pratiche e dinamiche per acquisire e sviluppare le competenze e le abilità che uno specialista deve sviluppare nel quadro della globalizzazione in cui viviamo.



#### Letture complementari

Articoli recenti, documenti di consenso e linee guida internazionali, tra gli altri. Nella biblioteca virtuale di TECH potrai accedere a tutto il materiale necessario per completare la tua specializzazione.





**Casi di Studio**

Completerai una selezione dei migliori casi di studio scelti appositamente per questo corso. Casi presentati, analizzati e monitorati dai migliori specialisti del panorama internazionale.



**Riepiloghi interattivi**

Il team di TECH presenta i contenuti in modo accattivante e dinamico in pillole multimediali che includono audio, video, immagini, diagrammi e mappe concettuali per consolidare la conoscenza.

Questo esclusivo sistema di specializzazione per la presentazione di contenuti multimediali è stato premiato da Microsoft come "Caso di successo in Europa".



**Testing & Retesting**

Valutiamo e rivalutiamo periodicamente le tue conoscenze durante tutto il programma con attività ed esercizi di valutazione e autovalutazione, affinché tu possa verificare come raggiungi progressivamente i tuoi obiettivi.



# 06 Titolo

Il Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza ti garantisce, oltre alla preparazione più rigorosa e aggiornata, l'accesso a una qualifica di Master Privato rilasciata da TECH Università Tecnologica.



“

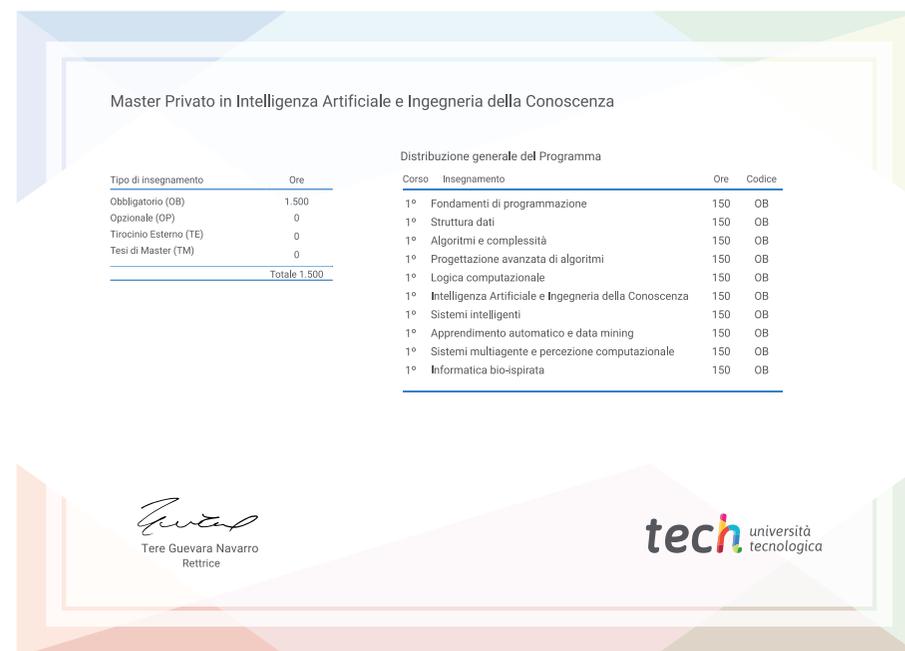
*Porta a termine questo programma e ricevi la tua qualifica universitaria senza spostamenti o fastidiose formalità”*

Questo **Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza** possiede il programma più completo e aggiornato del mercato.

Dopo aver superato la valutazione, lo studente riceverà mediante lettera certificata\* con ricevuta di ritorno, la sua corrispondente qualifica di **Master Privato** rilasciata da **TECH Università Tecnologica**.

Il titolo rilasciato da **TECH Università Tecnologica** esprime la qualifica ottenuta nel Master Privato, e riunisce tutti i requisiti comunemente richiesti da borse di lavoro, concorsi e commissioni di valutazione di carriere professionali.

Titolo: **Master Privato in Intelligenza Artificiale e Ingegneria della Conoscenza**  
N. Ore Ufficiali: **1.500 o.**



\*Se lo studente dovesse richiedere che il suo diploma cartaceo sia provvisto di Apostille dell'Aia, TECH EDUCATION effettuerà le gestioni opportune per ottenerla pagando un costo aggiuntivo.

futuro  
salute fiducia persone  
educazione informazione tutor  
garanzia accreditamento insegnamento  
istituzioni tecnologia apprendimento  
comunità impegno  
attenzione personalizzata innovazione  
conoscenza presente qualità  
formazione online  
sviluppo istituzioni  
classe virtuale lingu

**tech** università  
tecnologica

**Master Privato**  
Intelligenza Artificiale  
e Ingegneria della  
Conoscenza

- » Modalità: online
- » Durata: 12 mesi
- » Titolo: TECH Università Tecnologica
- » Dedizione: 16 ore/settimana
- » Orario: a scelta
- » Esami: online

# Master Privato

Intelligenza Artificiale e  
Ingegneria della Conoscenza