

# Master Privato

## Calcolo Parallelo e Distribuito





## Master Privato Calcolo Parallelo e Distribuito

- » 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/informatica/master/master-calcolo-parallelo-distribuito](http://www.techitute.com/it/informatica/master/master-calcolo-parallelo-distribuito)

# Indice

01

Presentazione

---

*pag. 4*

02

Obiettivi

---

*pag. 8*

03

Competenze

---

*pag. 14*

04

Direzione del corso

---

*pag. 18*

05

Struttura e contenuti

---

*pag. 22*

06

Metodologia

---

*pag. 32*

07

Titolo

---

*pag. 40*

# 01

# Presentazione

Oggi, la maggior parte dei software e dei sistemi elettronici utilizza, in un modo o nell'altro, il Calcolo Parallelo o Distribuito. Gli smartphone, ad esempio, hanno incorporato nel corso degli anni processori multi-core sempre più potenti, mentre per i computer moderni è ormai consuetudine impiegare processori multi-core. D'altra parte, l'informatica distribuita ha dato impulso ad altre branche della ricerca come i *big data*, essenziali in settori come le reti sociali, le reti aziendali o i giochi online multiplayer. Tutto ciò mostra l'importanza di queste due forme di programmazione, motivo per cui TECH ha creato questo programma universitario completo grazie al quale l'informatico conoscerà i vantaggi e i principali usi del calcolo parallelo e distribuito.





“

*Dai un impulso alla tua carriera e al tuo CV incorporando nel tuo lavoro le più recenti conoscenze in materia di Calcolo Parallelo e Distribuito”*

Una conoscenza avanzata del Calcolo Parallelo e Distribuito può dare una spinta alla carriera di qualsiasi informatico alla ricerca del miglior modo per distinguersi. Trattandosi di un argomento complesso che, oltretutto, può avere un gran varietà di utilizzi, TECH ha riunito un team di esperti del settore per prepararne i contenuti.

I professionisti troveranno quindi argomenti dedicati alla comunicazione e alla coordinazione nei sistemi informatici, all'analisi e alla programmazione di algoritmi paralleli o di sistemi distribuiti in informatica, oltre ad altri temi utili. Questi contenuti sono stati ideati da una prospettiva moderna e innovativa, basata sull'esperienza accumulata dal personale docente.

Pertanto, gli informatici che completino questo programma avranno un vantaggio decisivo per indirizzare la loro carriera verso lo sviluppo di applicazioni o sistemi nei settori del clima, della salute, dei *big data*, del cloud computing o della *blockchain*. Inoltre, data la natura avanzata del programma, sarà possibile pianificare un progetto di ricerca nel campo dell'informatica o in altre aree correlate.

Inoltre, il corso è offerto in un formato completamente online, eliminando la necessità di frequentare le lezioni o i vincoli tipici di un programma prestabilito. Gli studenti hanno la libertà di distribuire il carico di studio in base ai propri interessi, potendo combinare lo studio di questo Master Privato con altre responsabilità personali o professionali.

Questo **Master Privato in Calcolo Parallelo e Distribuito** possiede il programma più completo e aggiornato del mercato. Le caratteristiche principali del programma sono:

- ◆ Sviluppo di casi pratici presentati da esperti in Calcolo Parallelo e Distribuito
- ◆ Contenuti grafici, schematici ed eminentemente pratici che forniscono informazioni pratiche sulle discipline essenziali per l'esercizio della professione
- ◆ Esercizi pratici che offrono un processo di autovalutazione per migliorare l'apprendimento
- ◆ Speciale enfasi sulle metodologie innovative
- ◆ Lezioni teoriche, domande all'esperto e lavori di riflessione individuale
- ◆ Contenuti disponibili da qualsiasi dispositivo fisso o portatile provvisto di connessione a internet



*Iscriviti subito e scopri gli ultimi sviluppi del Calcolo Parallelo in ambienti cloud e della programmazione orientata al Calcolo Distribuito"*

“

*Sarai assistito in ogni momento dal team di docenti, composto da professionisti con una vasta esperienza nel campo del Calcolo Parallelo e Distribuito”*

Il personale docente del programma comprende rinomati professionisti del settore, nonché riconosciuti specialisti appartenenti a società scientifiche e università prestigiose, che forniscono agli studenti le competenze necessarie a intraprendere un percorso di studio eccellente.

I contenuti multimediali, sviluppati in base alle ultime tecnologie educative, forniranno al professionista un apprendimento coinvolgente e localizzato, ovvero inserito in un contesto reale.

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 accademico. Lo studente potrà usufruire di un innovativo sistema di video interattivi creati da esperti di rinomata fama.

*Studia nella più grande istituzione accademica online del mondo, TECH, che mette a tua disposizione la più recente tecnologia educativa.*

*Non farti scappare l'occasione di distinguerti e di dimostrare il tuo interesse per l'informatica di oggi e di domani.*



# 02 Obiettivi

Dato che l'informatica avanza a passi da gigante e che i professionisti che si dedicano a questo campo devono impegnarsi per aggiornare continuamente le proprie conoscenze, TECH ha incentrato questo programma sugli ultimi sviluppi in materia di Calcolo Parallelo e Distribuito. In questo modo, gli studenti non solo acquisiranno le competenze più avanzate in questo campo, ma impareranno anche a conoscere le molteplici applicazioni che tecnologie come la *blockchain* e il cloud computing hanno oggi.



“

*Avrai a portata di mano il tuo obiettivo di miglioramento professionale grazie ai suggerimenti e agli aspetti chiave sul calcolo che otterrai in questo programma”*



## Obiettivi generali

---

- ◆ Analizzare i diversi componenti del Calcolo Parallelo e Distribuito
- ◆ Misurare e confrontare le loro prestazioni per valutare il rendimento dell'insieme dei componenti utilizzati
- ◆ Analizzare in maniera approfondita il calcolo parallelo multiplatforma per utilizzare il parallelismo a livello di task tra diversi acceleratori hardware
- ◆ Analizzare in dettaglio il software e le architetture attuali
- ◆ Sviluppare in modo approfondito gli aspetti rilevanti del calcolo parallelo e distribuito
- ◆ Specializzare gli studenti nell'uso del calcolo parallelo e distribuito in diversi settori applicativi



## Obiettivi specifici

---

### Modulo 1. Parallelismo nel Calcolo Parallelo e Distribuito

- ◆ Analizzare i componenti di elaborazione: processore o memoria
- ◆ Approfondire l'architettura del parallelismo
- ◆ Analizzare le diverse forme di parallelismo dal punto di vista del processore

### Modulo 2. Suddivisione in parallelo nel Calcolo Parallelo e Distribuito

- ◆ Analizzare l'importanza della scomposizione dei processi paralleli nella risoluzione di problemi computazionali
- ◆ Esaminare diversi esempi per dimostrare l'applicazione e l'uso del calcolo e della sua scomposizione parallela
- ◆ Esporre procedure e strumenti che consentano l'esecuzione di processi paralleli, cercando di ottenere le migliori prestazioni possibili
- ◆ Sviluppare conoscenze specialistiche per identificare gli scenari di scomposizione dei processi paralleli e per scegliere e applicare la soluzione appropriata

### Modulo 3. Comunicazione e coordinamento nei sistemi di calcolo

- ◆ Analizzare le diverse architetture e i modelli di sistemi distribuiti
- ◆ Determinare le caratteristiche dei sistemi paralleli e distribuiti
- ◆ Approfondire le diverse comunicazioni che avvengono a livello dei processi
- ◆ Esaminare le comunicazioni remote, orientate al flusso, orientate ai messaggi e multicast, con esempi e considerazioni più recenti
- ◆ Stabilire i tipi di comunicazione che stanno emergendo, i loro punti di forza e i loro limiti

- ◆ Sviluppare i processi da seguire nella scelta degli algoritmi da applicare per il servizio di denominazione, la sincronizzazione dei timer, il coordinamento e l'accordo tra gli elementi del sistema
- ◆ Formulare scenari utilizzando diversi tipi di tecnologie di comunicazione che migliorino le prestazioni e la scalabilità

#### **Modulo 4. Analisi e programmazione di algoritmi paralleli**

- ◆ Analizzare i diversi paradigmi della programmazione parallela
- ◆ Esaminare gli strumenti più avanzati per la programmazione parallela
- ◆ Analizzare algoritmi paralleli per problemi fondamentali
- ◆ Concretizzare la progettazione e l'analisi di algoritmi paralleli
- ◆ Sviluppare algoritmi paralleli e implementarli utilizzando MPI, OpenMP, OpenCL/CUDA

#### **Modulo 5. Architetture parallele**

- ◆ Analizzare le principali architetture informatiche
- ◆ Approfondire aspetti chiave come il processo, il servizio e i thread di esecuzione
- ◆ Gestire i processi di esecuzione in un sistema operativo
- ◆ Utilizzare le classi per avviare e gestire i processi

#### **Modulo 6. Prestazioni in parallelo**

- ◆ Analizzare gli aspetti degli algoritmi paralleli che ne influenzano le prestazioni e la scalabilità
- ◆ Stabilire le principali metriche di prestazione e scalabilità degli algoritmi paralleli
- ◆ Esaminare le principali tecniche di confronto tra algoritmi paralleli
- ◆ Identificare i vincoli che le risorse hardware impongono alla parallelizzazione
- ◆ Determinare le migliori pratiche per prestazioni di programmi paralleli a memoria condivisa, a passaggio di messaggi, ibridi e con calcolo eterogeneo
- ◆ Compilare gli strumenti più avanzati per l'analisi delle prestazioni degli algoritmi paralleli
- ◆ Introdurre i principali modelli di elaborazione parallela
- ◆ Specificare una procedura robusta per la definizione di programmi paralleli ad alte prestazioni

#### **Modulo 7. Sistemi di Calcolo Distribuiti**

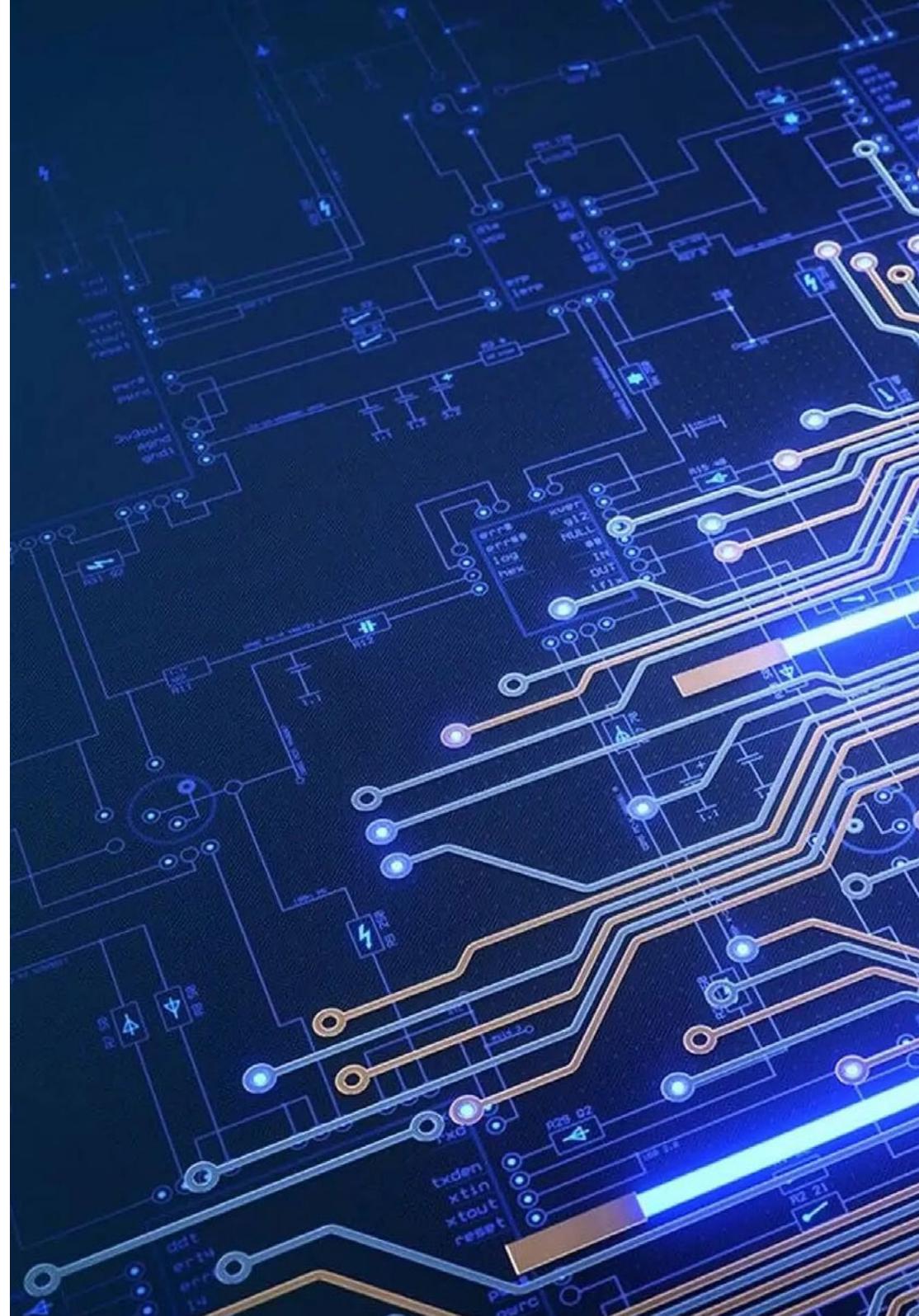
- ◆ Sviluppare gli elementi chiave di un sistema distribuito
- ◆ Esaminare gli elementi di sicurezza applicati nei sistemi distribuiti e la loro necessità
- ◆ Presentare i diversi tipi di sistemi distribuiti più comunemente utilizzati, caratteristiche, funzionalità e problemi da risolvere
- ◆ Dimostrare il teorema PAC applicabile ai sistemi distribuiti: *Consistency, Availability e Partition Tolerance*

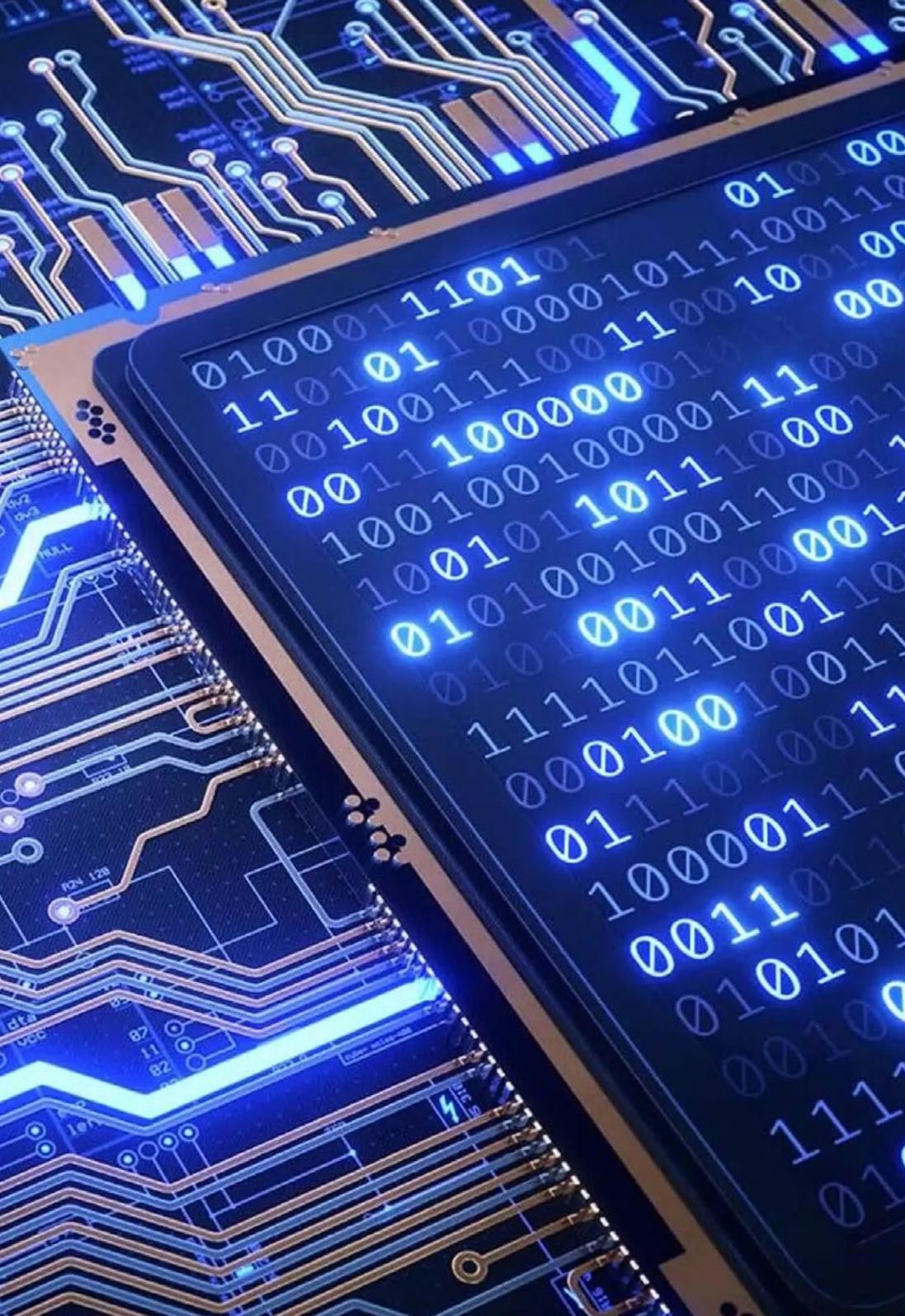
### Modulo 8. Calcolo Parallelo applicato agli ambienti Cloud

- ◆ Sviluppare il paradigma del cloud computing
- ◆ Identificare diversi approcci in base al grado di automazione e di servizio
- ◆ Analizzare i principali elementi di un'architettura cloud
- ◆ Stabilire le differenze con un'architettura *on-premise*
- ◆ Analizzare le diverse opzioni di deployment cloud: *Multi-cloud, Hybrid Cloud*
- ◆ Approfondire i vantaggi intrinseci del cloud computing
- ◆ Sviluppare i principi economici del cloud computing: passare da CAPEX a OPEX
- ◆ Valutare l'offerta commerciale di diversi fornitori cloud
- ◆ Valutare le capacità di supercalcolo nel cloud
- ◆ Esaminare la sicurezza del cloud computing

### Modulo 9. Modelli e semantica formale. Programmazione orientata al Calcolo Distribuito

- ◆ Identificare i vantaggi della semantica formale
- ◆ Esaminare come la semantica formale aiuti la programmazione orientata al calcolo distribuito
- ◆ Concretizzare le possibilità della semantica formale applicata alla programmazione orientata al calcolo distribuito
- ◆ Sviluppare in modo approfondito i principali strumenti in termini di fattibilità dei progetti che utilizzano questa tecnologia
- ◆ Identificare i linguaggi di programmazione nel modello semantico
- ◆ Determinare come questi modelli semantici incidano sui linguaggi di programmazione
- ◆ Valutare e confrontare i modelli computazionali
- ◆ Concretizzare l'uso di modelli distribuiti
- ◆ Presentare gli strumenti di mercato più avanzati da impiegare nei progetti





### Modulo 10. Applicazioni del Calcolo Parallelo e Distribuito

- ◆ Dimostrare il gran contributo delle applicazioni del calcolo parallelo e distribuito al nostro settore
- ◆ Determinare le architetture rilevanti sul mercato
- ◆ Valutare i benefici di questi casi d'uso
- ◆ Presentare soluzioni di successo sul mercato
- ◆ Dimostrare perché è importante valutare il cambiamento climatico
- ◆ Determinare l'importanza della GPU oggi
- ◆ Presentare l'impatto di questa tecnologia sulle reti elettriche
- ◆ Esplorare motori distribuiti per offrire un servizio ai nostri clienti
- ◆ Conoscere i vantaggi dei motori distribuiti per ottenere profitti per le nostre imprese
- ◆ Presentare esempi di database in memoria e la loro importanza
- ◆ Esaminare come questi modelli incidano nel settore della medicina

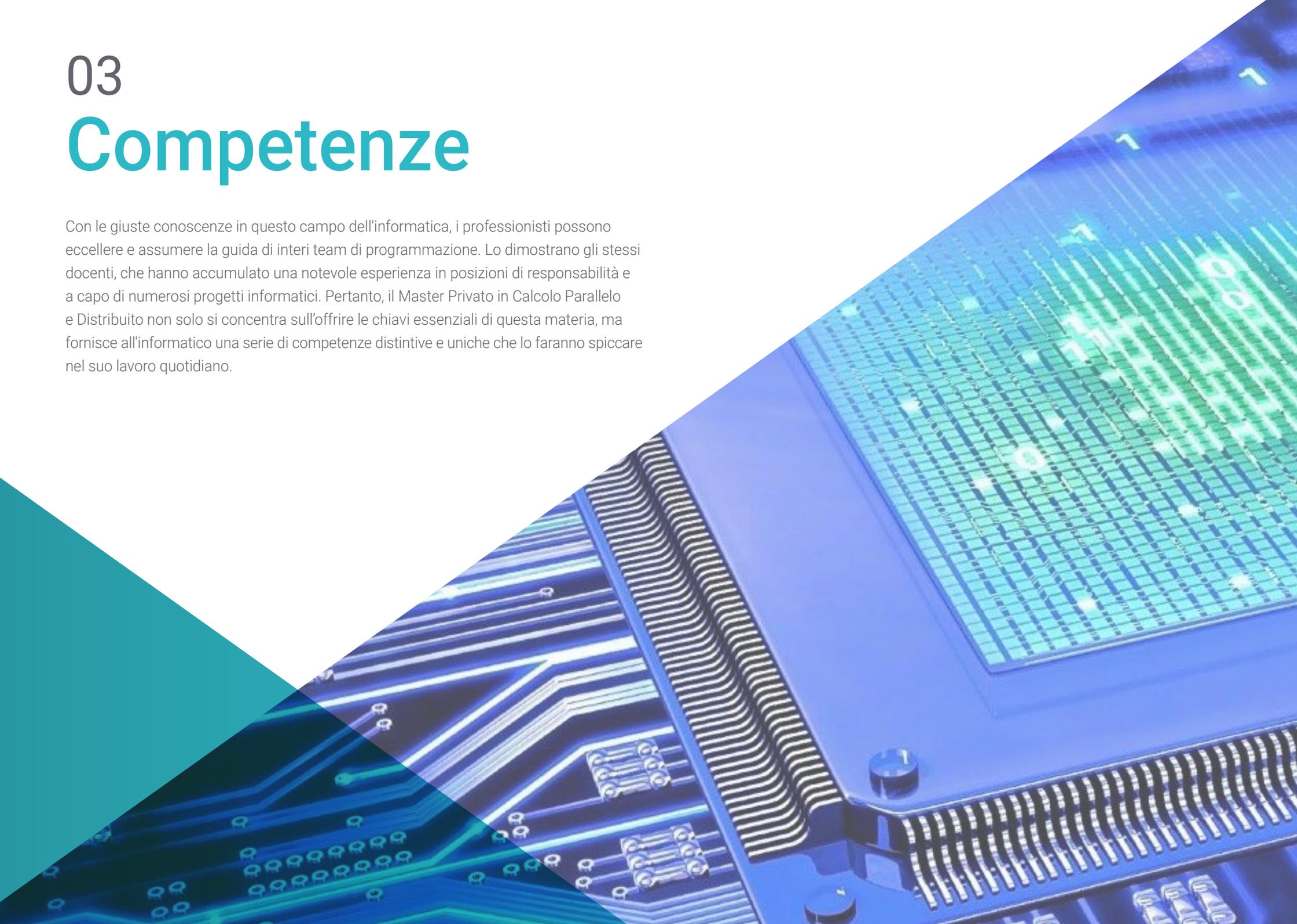
“

*Otterrai una panoramica completa su tutte le questioni più importanti del Calcolo Parallelo e Distribuito, sulla base dei parallelismi propri delle numerose applicazioni che ne derivano"*

# 03

## Competenze

Con le giuste conoscenze in questo campo dell'informatica, i professionisti possono eccellere e assumere la guida di interi team di programmazione. Lo dimostrano gli stessi docenti, che hanno accumulato una notevole esperienza in posizioni di responsabilità e a capo di numerosi progetti informatici. Pertanto, il Master Privato in Calcolo Parallelo e Distribuito non solo si concentra sull'offrire le chiavi essenziali di questa materia, ma fornisce all'informatico una serie di competenze distintive e uniche che lo faranno spiccare nel suo lavoro quotidiano.



“

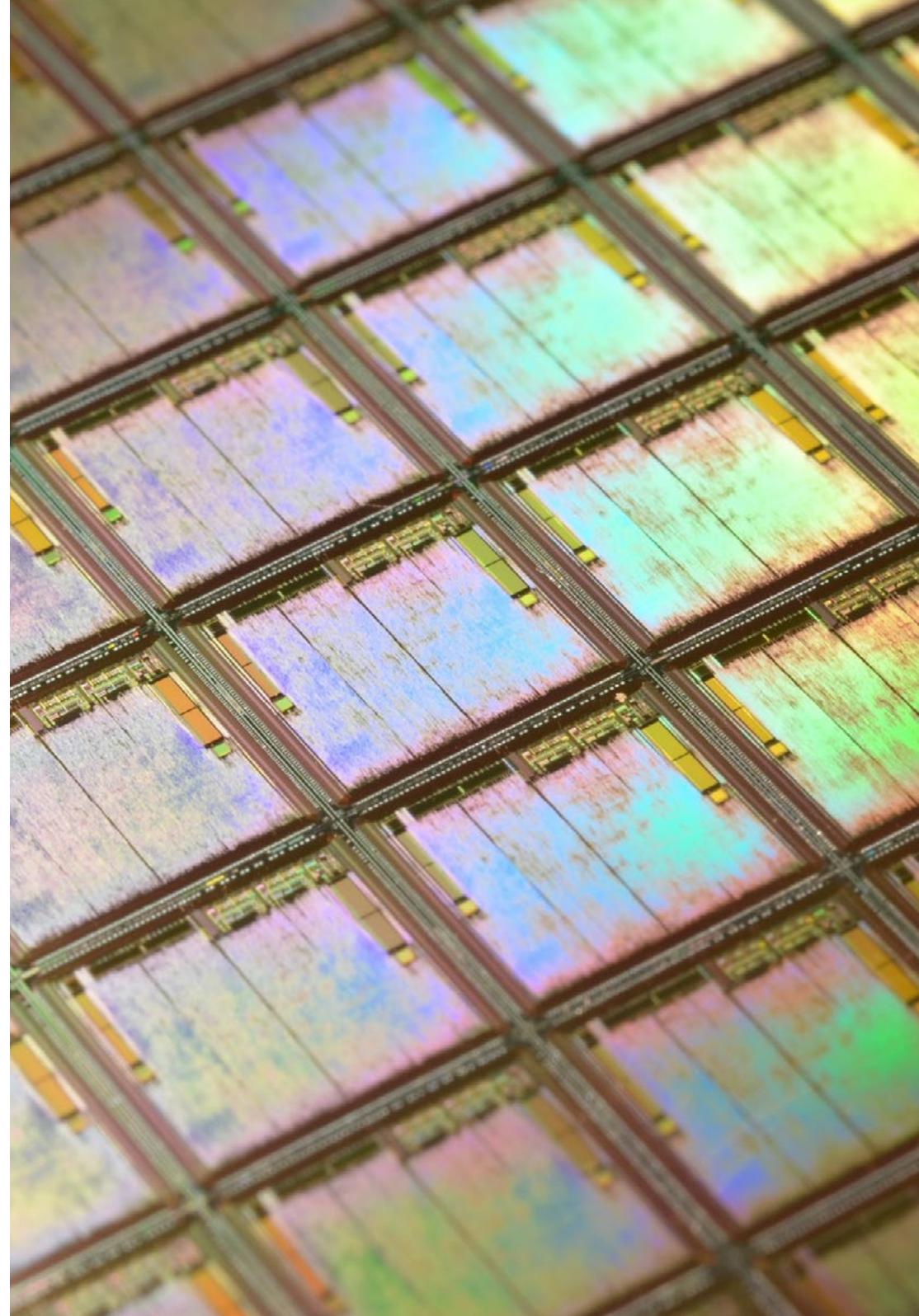
*Questa qualifica sarà la chiave per una rapida ascesa verso posizioni di responsabilità nella programmazione e nello sviluppo di ogni tipo di software"*



## Competenze generali

---

- ◆ Sviluppare conoscenze specializzate sui diversi tipi di parallelismo
- ◆ Analizzare una strategia di parallelizzazione basata su metriche di prestazione
- ◆ Determinare le caratteristiche principali del calcolo parallelo e distribuito prima di affrontare la comunicazione e il coordinamento che si verifica tra i suoi componenti
- ◆ Dimostrare che in questi sistemi possono avvenire comunicazioni interprocesso, chiamate remote o comunicazioni indirette
- ◆ Determinare gli aspetti che danneggiano le prestazioni delle applicazioni parallele
- ◆ Analizzare le tecniche avanzate per ottimizzare il codice parallelo e la comunicazione nei sistemi a memoria distribuita, il controllo dell'affinità, il bilanciamento del carico e la gestione parallela di input/output.
- ◆ Esaminare i modelli di programmazione ibrida per i sistemi con più acceleratori hardware e i modelli di programmazione ibrida per i sistemi con memoria condivisa/distribuita





## Competenze specifiche

---

- ◆ Determinare i problemi di scalabilità e di prestazioni che possono essere risolti con la scomposizione dei processi paralleli
- ◆ Analizzare le caratteristiche di una proposta di parallelismo a memoria condivisa, di parallelismo a trasmissione di messaggi e di parallelismo su GPU, nonché lo scenario ibrido
- ◆ Stabilire la necessità della resilienza dei processi e della modellazione degli stakeholder nella risoluzione degli attuali problemi informatici
- ◆ Fornire esempi o casi in cui la scomposizione parallela ha avuto un'applicazione di successo in termini di prestazioni o scalabilità
- ◆ Analizzare e progettare algoritmi paralleli
- ◆ Compilazione delle primitive principali di MPI, OpenMP, OpenCL/CUDA
- ◆ Esaminare i processi vettoriali e matriciali
- ◆ Analizzare la programmazione parallela e distribuita, i linguaggi di programmazione, gli strumenti e i modelli di progettazione
- ◆ Identificare gli elementi che consentono l'interconnessione delle reti distribuite
- ◆ Guidare passo a passo la progettazione di un sistema distribuito
- ◆ Valutare i diversi tipi di replica dei dati nei sistemi esistenti
- ◆ Compilare gli approcci pratici alla sicurezza applicabili

# 04

## Direzione del corso

Nella sua incessante ricerca della massima qualità didattica possibile, TECH ha riunito in questo programma un gruppo di ingegneri informatici con esperienza internazionale in una gran varietà di progetti legati al Calcolo Parallelo e Distribuito. Gli studenti acquisiranno pertanto conoscenze che non sono focalizzate solo sul loro miglioramento professionale, ma che sono basate anche sull'esperienza dei docenti nella gestione di team informatici multidisciplinari.



“

*Sviluppa il tuo potenziale accademico e professionale grazie ai migliori insegnanti, che si impegneranno a stare al tuo fianco per tutta la durata del corso"*

## Direzione



### Dott. Olalla Bonal, Martín

- ◆ Responsabile senior della pratica Blockchain presso EY
- ◆ Specialista tecnico Blockchain Client presso IBM
- ◆ Direttore dell'architettura presso Blocknitive
- ◆ Coordinatore del Team di Database distribuiti non relazionali presso wedoIT (filiale di IBM)
- ◆ Architetto di Infrastrutture presso Bankia
- ◆ Responsabile del reparto layout presso T-Systems
- ◆ Coordinatore del dipartimento di Bing Data España S.L.

## Personale docentet

### Dott. Villot Guisán, Pablo

- ◆ Chief Information Officer, Chief Technical Officer e Fondatore di New Tech & Talent
- ◆ Esperto di tecnologia presso KPMG Spagna
- ◆ Architetto *Blockchain* in Everis
- ◆ Sviluppatore J2EE Area Logistica Commerciale presso Inditex
- ◆ Laurea in Ingegneria informatica presso l'Università di La Coruña
- ◆ Certificazione Microsoft MSCA: *Cloud Platform*

### Dott. Blanco, Eduardo

- ◆ Specializzazione in Informatica
- ◆ Docente presso l'Università Simón Bolívar
- ◆ Dottorato in Informatica presso l'Università Simón Bolívar
- ◆ Ingegnere in Informatica presso l'Università Simón Bolívar
- ◆ Master in Informatica presso l'Università Simón Bolívar

**Dott. Almendras Aruzamen, Luis Fernando**

- ◆ Ingegnere di Dati e Business Intelligence. Gruppo Solutia, Madrid
- ◆ Ingegnere dei dati presso Indizen
- ◆ Ingegnere dei dati e della *business intelligence* in Tecnologia e Persone
- ◆ Ingegnere di supporto per database, *big data* e *business intelligence* presso Equinix
- ◆ Ingegnere dei Dati. Jalasoft
- ◆ Product Manager e responsabile dell'analisi aziendale presso Goja
- ◆ Vicedirettore Business Intelligence. VIVA Nuevatel PC's
- ◆ Responsabile dell'area data warehouse e big data di Viva
- ◆ Leader dello sviluppo software presso Intersoft
- ◆ Laurea in Informatica conseguita presso l'Università Superiore di San Simón
- ◆ Dottorato in Ingegneria informatica. Università Complutense di Madrid
- ◆ Master in Ingegneria Informatica presso l'Università Complutense di Madrid
- ◆ Master in Sistemi Informativi e Gestione Tecnologica presso l'Università Superiore di San Simón
- ◆ Istruttore internazionale: Oracle Database. Proydesa- Oracle, Argentina
- ◆ Certificazione professionale di Project Management. Outreach Consulting, Cile

**Dott.ssa Carratalá Sáez, Rocío**

- ◆ Ricercatrice specializzata in Informatica
- ◆ Docente di studi universitari legati all'informatica
- ◆ Dottorato di ricerca in Informatica conseguito presso l'Università Jaume I
- ◆ Dottorato di ricerca in Matematica computazionale conseguito presso l'Università Jaume I
- ◆ Master in Calcolo Parallelo e Distribuito presso l'Università Politecnica di Valencia
- ◆ Corsi di specializzazione legati all'informatica, alla matematica e agli strumenti di ricerca accademica

**Dott. Gozalo Fernández, Juan Luis**

- ◆ Responsabile dei prodotti basati su Blockchain presso Open Canarias
- ◆ Direttore Blockchain DevOps presso Alastria
- ◆ Direttore della tecnologia a livello di servizio presso Santander Spagna
- ◆ Responsabile dello Sviluppo dell'Applicazione Mobile di Tinkerlink presso Cronos Telecom
- ◆ Direttore della Tecnologia di Gestione dei Servizi IT presso Barclays Bank Spagna
- ◆ Laurea in Ingegneria Informatica presso l'UNED
- ◆ Specializzazione in *Deep Learning* presso DeepLearning.ai

**Dott. Gómez Gómez, Borja**

- ◆ Responsabile dello Sviluppo Aziendale per l'Innovazione del Cloud presso Oracle
- ◆ Responsabile Blockchain e soluzioni di architettura prevendita presso Paradigma Digital
- ◆ Architetto IT Consulente senior presso Atmira
- ◆ Architetto SOA e consulente TCP SI
- ◆ Analista e consulente presso Everis
- ◆ Laurea in Ingegneria Informatica presso l'Università Complutense di Madrid
- ◆ Master in Science Computer Engineering presso l'Università Complutense di Madrid

# 05

## Struttura e contenuti

Al fine di facilitare lo studio e l'acquisizione di competenze, TECH ha incorporato la metodologia pedagogica più efficace in questo programma universitario. Grazie al *Relearning*, gli studenti potranno ridurre notevolmente il tempo per acquisire le conoscenze più importanti del programma, ulteriormente rafforzato dalla grande quantità di materiale audiovisivo, di letture complementari e di esercizi pratici che aiutano a consolidare l'intera materia.



“

*Esercizi pratici basati su casi reali e video dettagliati elaborati dagli stessi docenti saranno la chiave del tuo successo in questo programma universitario"*

## Modulo 1. Parallelismo nel Calcolo Parallelo e Distribuito

- 1.1. Elaborazione parallela
  - 1.1.1. Elaborazione parallela
  - 1.1.2. Elaborazione parallela nel calcolo. Scopi
  - 1.1.3. Elaborazione parallela. Analisi
- 1.2. Sistemi paralleli
  - 1.2.1. Sistemi paralleli
  - 1.2.2. Livelli di parallelismo
  - 1.2.3. Composizione del sistema parallelo
- 1.3. Architettura dei processori
  - 1.3.1. Complessità del processore
  - 1.3.2. Architettura del processore. Modalità di funzionamento
  - 1.3.3. Architettura del processore. Organizzazione della memoria
- 1.4. Reti nell'elaborazione parallela
  - 1.4.1. Modalità di funzionamento
  - 1.4.2. Strategia di controllo
  - 1.4.3. Tecniche di commutazione
  - 1.4.4. Topologia
- 1.5. Architetture parallele
  - 1.5.1. Algoritmi
  - 1.5.2. Collegamenti
  - 1.5.3. Comunicazione
- 1.6. Prestazioni del calcolo parallelo
  - 1.6.1. Evoluzione del rendimento
  - 1.6.2. *Performance*
  - 1.6.3. Calcolo parallelo Casi di studio
- 1.7. Tassonomia Flynn
  - 1.7.1. MIMD: memoria condivisa
  - 1.7.2. MIMD: memoria distribuita
  - 1.7.3. MIMD: sistemi ibridi
  - 1.7.4. Flusso di dati

- 1.8. Forme di parallelismo: TLP (*Thread Level Parallelism*)
  - 1.8.1. Forme di parallelismo: TLP (*Thread Level Parallelism*)
  - 1.8.2. *Coarse grain*
  - 1.8.3. *Fine grain*
  - 1.8.4. SMT
- 1.9. Forme di parallelismo: DLP (*Date Level Parallelism*)
  - 1.9.1. Forme di parallelismo: DLP (*Date Level Parallelism*)
  - 1.9.2. *Short vector processing*
  - 1.9.3. *Vector processors*
- 1.10. Forme di parallelismo: ILP (*Date Level Parallelism*)
  - 1.10.1. Forme di parallelismo: ILP (*Date Level Parallelism*)
  - 1.10.2. Processori segmentati
  - 1.10.3. Processore superscalare
  - 1.10.4. Processore *Very Long Instruction Word* (VLIW)

## Modulo 2. Suddivisione in parallelo nel Calcolo Parallelo e Distribuito

- 2.1. Scomposizione parallela
  - 2.1.1. Elaborazione parallela
  - 2.1.2. Architetture
  - 2.1.3. Supercomputer
- 2.2. Hardware e software parallelo
  - 2.2.1. Sistemi seriali
  - 2.2.2. Hardware parallelo
  - 2.2.3. Software parallelo
  - 2.2.4. Input e output
  - 2.2.5. Prestazioni
- 2.3. Scalabilità parallela e problemi di prestazioni ricorrenti
  - 2.3.1. Parallelismo
  - 2.3.2. Scalabilità parallela
  - 2.3.3. Problemi comuni di prestazioni

- 2.4. Parallelismo della memoria condivisa
  - 2.4.1. Parallelismo della memoria condivisa
  - 2.4.2. OpenMP e Pthreads
  - 2.4.3. Parallelismo della memoria condivisa. Esempi
- 2.5. Unità di elaborazione grafica (GPU)
  - 2.5.1. Unità di elaborazione grafica (GPU)
  - 2.5.2. Architettura di dispositivi unificati per il calcolo (CUDA)
  - 2.5.3. Architettura di dispositivi unificati per il calcolo. Esempi
- 2.6. Sistemi di trasmissione di messaggi
  - 2.6.1. Sistemi di trasmissione di messaggi
  - 2.6.2. MPI. Interfaccia di trasmissione di messaggi
  - 2.6.3. Sistemi di trasmissione di messaggi. Esempi
- 2.7. Parallelizzazione ibrida con MPI e OpenMP
  - 2.7.1. Programmazione ibrida
  - 2.7.2. Modelli di programmazione MPI/OpenMP
  - 2.7.3. Decomposizione e mappatura ibrida
- 2.8. Calcolo MapReduce
  - 2.8.1. Hadoop
  - 2.8.2. Altri sistemi di gestione
  - 2.8.3. Calcolo parallelo. Esempi
- 2.9. Modello degli stakeholder e processi reattivi
  - 2.9.1. Modello degli stakeholder
  - 2.9.2. Processi reattivi
  - 2.9.3. Stakeholder e processi reattivi. Esempi
- 2.10. Scenari del calcolo parallelo
  - 2.10.1. Elaborazione audio di immagini
  - 2.10.2. Statistica/Data Mining
  - 2.10.3. Gestione parallela
  - 2.10.4. Operazioni matriciali parallele

### Modulo 3. Comunicazione e coordinamento nei sistemi di calcolo

- 3.1. Processi di Calcolo Parallelo e Distribuito
  - 3.1.1. Processi di Calcolo Parallelo e Distribuito
  - 3.1.2. Processi e fili
  - 3.1.3. Virtualizzazione
  - 3.1.4. Clienti e server
- 3.2. Comunicazione nel calcolo parallelo
  - 3.2.1. Comunicazione nel calcolo parallelo
  - 3.2.2. Protocolli a strati
  - 3.2.3. Comunicazione nel calcolo parallelo. Tipologia
- 3.3. Chiamata di procedura remota
  - 3.3.1. Funzionamento di RPC (*Remote Procedure Call*)
  - 3.3.2. Passaggio di parametri
  - 3.3.3. RPC asincrono
  - 3.3.4. Procedura remota. Esempi
- 3.4. Comunicazione orientata ai messaggi
  - 3.4.1. Comunicazione transitoria orientata ai messaggi
  - 3.4.2. Comunicazione persistente orientata ai messaggi
  - 3.4.3. Comunicazione orientata ai messaggi. Esempi
- 3.5. Comunicazione orientata ai flussi
  - 3.5.1. Supporto per mezzi continui
  - 3.5.2. Flussi e qualità del servizio
  - 3.5.3. Sincronizzazione dei flussi
  - 3.5.4. Comunicazione orientata ai flussi. Esempi
- 3.6. Comunicazione multicast
  - 3.6.1. Multicast a livello di applicazione
  - 3.6.2. Diffusione dei dati basata su voci
  - 3.6.3. Comunicazione multicast. Esempi

- 3.7. Altri tipi di comunicazione
  - 3.7.1. Invocazione di un metodo remoto
  - 3.7.2. Servizi Web / SOA / REST
  - 3.7.3. Notifica degli eventi
  - 3.7.4. Agenti mobili
- 3.8. Servizio nominali
  - 3.8.1. Servizi nominali nel calcolo
  - 3.8.2. Servizi nominali e sistema di nomi di dominio
  - 3.8.3. Servizi di directory
- 3.9. Sincronizzazione
  - 3.9.1. Sincronizzazione degli orologi
  - 3.9.2. Orologi logici, mutua esclusione e posizionamento globale dei nodi
  - 3.9.3. Scelta degli algoritmi
- 3.10. Comunicazione. Coordinamento e accordo
  - 3.10.1. Coordinamento e accordo
  - 3.10.2. Coordinamento e accordo. Consenso e problemi
  - 3.10.3. Comunicazione e coordinamento. Attualità

## Modulo 4. Analisi e la programmazione di algoritmi paralleli

- 4.1. Algoritmi paralleli
  - 4.1.1. Scomposizione del problema
  - 4.1.2. Dipendenze dai dati
  - 4.1.3. Parallelismo implicito ed esplicito
- 4.2. Paradigma di programmazione parallela
  - 4.2.1. Programmazione parallela con memoria condivisa
  - 4.2.2. Programmazione parallela con memoria distribuita
  - 4.2.3. Programmazione parallela ibrida
  - 4.2.4. Calcolo eterogeneo - CPU + GPU
  - 4.2.5. Calcolo quantistico. Nuovi modelli di programmazione con parallelismo implicito

- 4.3. Programmazione parallela con memoria condivisa
  - 4.3.1. Modelli di programmazione parallela con memoria condivisa
  - 4.3.2. Algoritmi paralleli con memoria condivisa
  - 4.3.3. Librerie per programmazione parallela con memoria condivisa
- 4.4. OpenMP
  - 4.4.1. OpenMP
  - 4.4.2. Esecuzione e debug dei programmi OpenMP
  - 4.4.3. Algoritmi paralleli con memoria condivisa OpenMP
- 4.5. Programmazione parallela tramite condivisione di messaggi
  - 4.5.1. Primitive di trasmissione di messaggi
  - 4.5.2. Operazioni di calcolo e comunicazione collettiva
  - 4.5.3. Algoritmi paralleli tramite condivisione di messaggi
  - 4.5.4. Librerie per programmazione parallela con condivisione di messaggi
- 4.6. *Message Passing Interface* (MPI)
  - 4.6.1. *Message Passing Interface* (MPI)
  - 4.6.2. Esecuzione e debug dei programmi OpenMP
  - 4.6.3. Algoritmi paralleli tramite condivisione di messaggi con MPI
- 4.7. Programmazione parallela ibrida
  - 4.7.1. Programmazione parallela ibrida
  - 4.7.2. Esecuzione e debug dei programmi ibridi paralleli
  - 4.7.3. Algoritmi paralleli ibridi MPI-OpenMP
- 4.8. Programmazione parallela con calcolo eterogeneo
  - 4.8.1. Programmazione parallela con calcolo eterogeneo
  - 4.8.2. CPU vs. GPU
  - 4.8.3. Algoritmi paralleli con calcolo eterogeneo
- 4.9. OpenCL e CUDA
  - 4.9.1. OpenCL vs. CUDA
  - 4.9.2. Esecuzione e debug dei programmi con calcolo eterogeneo paralleli
  - 4.9.3. Algoritmi paralleli con calcolo eterogeneo

- 4.10. Progettazione degli algoritmi paralleli
  - 4.10.1. Progettazione degli algoritmi paralleli
  - 4.10.2. Problema e contesto
  - 4.10.3. Parallellizzazione automatica vs. Parallellizzazione manuale
  - 4.10.4. Suddivisione del problema
  - 4.10.5. Comunicazioni nel calcolo

## Modulo 5. Architetture parallele

- 5.1. Architetture parallele
  - 5.1.1. Sistemi paralleli. Classificazione
  - 5.1.2. Fonti di parallelismo
  - 5.1.3. Parallelismo e processori
- 5.2. Prestazioni dei sistemi paralleli
  - 5.2.1. Metriche e misure di performance
  - 5.2.2. *Speed-up*
  - 5.2.3. Granularità dei sistemi paralleli
- 5.3. Processori vettoriali
  - 5.3.1. Processore vettoriale di base
  - 5.3.2. Memoria interfogliata
  - 5.3.3. Prestazioni dei processori vettoriali
- 5.4. Processori a matrice
  - 5.4.1. Organizzazione di base
  - 5.4.2. Programmazione di processori matriciali
  - 5.4.3. Programmazione di processori matriciali. Esempio pratico
- 5.5. Reti di interconnessione
  - 5.5.1. Reti di interconnessione
  - 5.5.2. Topologia, controllo del flusso e routing
  - 5.5.3. Reti di interconnessione. Classificazione in base alla topologia
- 5.6. Multiprocessori
  - 5.6.1. Reti di interconnessione per multiprocessori
  - 5.6.2. Coerenza della memoria e della cache
  - 5.6.3. Protocolli di sondaggio
- 5.7. Sincronizzazione
  - 5.7.1. Serrature (esclusione reciproca)
  - 5.7.2. Eventi di sincronizzazione P2P
  - 5.7.3. Eventi di sincronizzazione globale
- 5.8. Multicomputer
  - 5.8.1. Reti di interconnessione per multicomputer
  - 5.8.2. Livello di commutazione
  - 5.8.3. Livello di routing
- 5.9. Architetture avanzate
  - 5.9.1. Macchine dei flussi di dati
  - 5.9.2. Altre architetture
- 5.10. Programmazione parallela e distribuita
  - 5.10.1. Linguaggi di programmazione parallela
  - 5.10.2. Strumenti di programmazione parallela
  - 5.10.3. Modelli di design
  - 5.10.4. Concorrenza dei linguaggi di programmazione paralleli e distribuiti

## Modulo 6. Prestazioni in parallelo

- 6.1. Performance degli algoritmi paralleli
  - 6.1.1. Legge di Ahmdal
  - 6.1.2. Legge di Gustafson
  - 6.1.3. Metriche di prestazione e scalabilità degli algoritmi paralleli
- 6.2. Confronto degli algoritmi paralleli
  - 6.2.1. *Benchmarking*
  - 6.2.2. Analisi matematica di algoritmi paralleli
  - 6.2.3. Analisi matematica di algoritmi paralleli
- 6.3. Restrizioni delle risorse hardware
  - 6.3.1. Memoria
  - 6.3.2. Processo
  - 6.3.3. Comunicazioni
  - 6.3.4. Partizione dinamica delle risorse
- 6.4. Prestazioni dei programmi paralleli con memoria condivisa
  - 6.4.1. Suddivisione ottimale dei compiti
  - 6.4.2. Affinità di *Threads*
  - 6.4.3. Parallelismo SIMD
  - 6.4.4. Programmi paralleli con memoria condivisa. Esempi
- 6.5. Prestazioni dei programmi paralleli tramite condivisione di messaggi
  - 6.5.1. Prestazioni dei programmi paralleli tramite passaggio di messaggi
  - 6.5.2. Ottimizzazione di comunicazione in MPI
  - 6.5.3. Controllo di affinità e bilanciamento del carico
  - 6.5.4. I/O Parallela
  - 6.5.5. Programmi paralleli tramite condivisione di messaggi. Esempi
- 6.6. Prestazioni dei programmi ibridi paralleli
  - 6.6.1. Prestazioni dei programmi ibridi paralleli
  - 6.6.2. Programmazione ibrida per sistemi a memoria condivisa/distribuita
  - 6.6.3. Programmi paralleli ibridi. Esempi

- 6.7. Prestazioni dei programmi con elaborazione eterogenea
  - 6.7.1. Prestazioni dei programmi con elaborazione eterogenea
  - 6.7.2. Programmazione ibrida per sistemi con più acceleratori hardware
  - 6.7.3. Programmi con calcolo eterogeneo. Esempi
- 6.8. Analisi di prestazione di algoritmi paralleli
  - 6.8.1. Analisi di prestazione di algoritmi paralleli
  - 6.8.2. Analisi di prestazione di algoritmi paralleli. Strumenti
  - 6.8.3. Analisi di prestazione di algoritmi paralleli. Raccomandazioni
- 6.9. Modelli paralleli
  - 6.9.1. Modelli paralleli
  - 6.9.2. Principali modelli paralleli
  - 6.9.3. Modelli paralleli. Confronto
- 6.10. Programmi paralleli ad alte prestazioni
  - 6.10.1. Processo
  - 6.10.2. Programmi paralleli ad alte prestazioni
  - 6.10.3. Programmi paralleli ad alte prestazioni. Casi pratici

## Modulo 7. Sistemi di Calcolo Distribuiti

- 7.1. Sistemi Distribuiti
  - 7.1.1. Sistemi distribuiti
  - 7.1.2. Dimostrazione del teorema di PAC (o Congettura di Brewer)
  - 7.1.3. Errori di programmazione sui sistemi distribuiti
  - 7.1.4. Computazione ubiqua
- 7.2. Sistemi distribuiti. Caratteristiche
  - 7.2.1. Eterogeneità
  - 7.2.2. Estensibilità
  - 7.2.3. Sicurezza
  - 7.2.4. Scalabilità
  - 7.2.5. Tolleranza agli errori
  - 7.2.6. Concorrenza
  - 7.2.7. Trasparenza

- 7.3. Reti e interconnessioni di reti distribuite
  - 7.3.1. Reti e sistemi distribuiti. Le prestazioni delle reti
  - 7.3.2. Reti disponibili per creare un sistema distribuito. Tipologia
  - 7.3.3. Protocolli di rete distribuiti vs. centralizzati
  - 7.3.4. Interconnessione di reti. Internet
- 7.4. Comunicazione tra processi distribuiti
  - 7.4.1. Comunicazione tra nodi di un sistema distribuito. Problemi e difetti
  - 7.4.2. Meccanismi da implementare su RPC e RDMA per evitare guasti
  - 7.4.3. Meccanismi da implementare nei software per evitare guasti
- 7.5. Progettazione di sistemi distribuiti
  - 7.5.1. Progettazione efficiente di sistemi distribuiti
  - 7.5.2. Schemi per la programmazione in sistemi distribuiti (SD)
  - 7.5.3. Architettura Orientata ai Servizi [*Service Oriented Architecture (SOA)*]
  - 7.5.4. *Service Orchestration e Microservices Data Management*
- 7.6. Operazione di sistemi distribuiti
  - 7.6.1. Monitoraggio dei sistemi
  - 7.6.2. Implementazione di un efficiente Trace System (*Logging*) in un SD
  - 7.6.3. Monitoraggio nelle reti distribuite
  - 7.6.4. Utilizzo di uno strumento di monitoraggio per una scheda SD Prometheus e Grafana
- 7.7. Replica dei sistemi
  - 7.7.1. Replica dei sistemi. Tipologie
  - 7.7.2. Architetture immutabili
  - 7.7.3. Sistemi Container e virtualizzatori come Sistemi Distribuiti
  - 7.7.4. Reti *Blockchain* come Sistemi Distribuiti
- 7.8. Sistemi distribuiti multimediali
  - 7.8.1. Condivisione distribuita di immagini e video. Problema
  - 7.8.2. Server di oggetti multimediali
  - 7.8.3. Topologia di rete per un sistema multimediale
  - 7.8.4. Analisi dei sistemi multimediali distribuiti: Netflix, Amazon, Spotify, ecc.
  - 7.8.5. I sistemi multimediali distribuiti nell'istruzione

- 7.9. File system distribuiti
  - 7.9.1. Scambio distribuito di file. Problema
  - 7.9.2. Applicabilità del teorema di PAC alle banche dati
  - 7.9.3. File system web distribuiti: Akamai
  - 7.9.4. File system documentari distribuiti IPFS
  - 7.9.5. Sistemi di database distribuiti
- 7.10. Approcci alla sicurezza nei sistemi distribuiti
  - 7.10.1. Sicurezza in Sistemi Distribuiti
  - 7.10.2. Attacchi noti ai sistemi distribuiti
  - 7.10.3. Strumenti per testare la sicurezza di una SD

## Modulo 8. Calcolo Parallelo Applicato agli Ambienti *Cloud*

- 8.1. Cloud computing
  - 8.1.1. Stato dell'arte del panorama IT
  - 8.1.2. Il cloud
  - 8.1.3. Cloud computing
- 8.2. Sicurezza e resilienza del cloud
  - 8.2.1. Regioni, disponibilità e zone di guasto
  - 8.2.2. Amministrazione dei *tenant* o account di *cloud*
  - 8.2.3. Controllo dell'identità e dell'accesso nel cloud
- 8.3. *Networking* nel cloud
  - 8.3.1. Reti virtuali definite dal software
  - 8.3.2. Componenti di rete di un network definito dal software
  - 8.3.3. Collegamento con altri sistemi
- 8.4. Servizi Cloud
  - 8.4.1. Infrastruttura come servizio
  - 8.4.2. Piattaforma come servizio
  - 8.4.3. Computing *serverless*
  - 8.4.4. Software come servizio

- 8.5. Archiviazione in cloud
  - 8.5.1. Archiviazione dei blocchi nel cloud
  - 8.5.2. Salvataggio dei fascicoli nel cloud
  - 8.5.3. Salvataggio degli obiettivi nel cloud
- 8.6. Interazioni e monitoraggio del cloud
  - 8.6.1. Monitoraggio e gestione del cloud
  - 8.6.2. Interazione con il cloud: console di gestione
  - 8.6.3. Interazione con *Command Line Interface*
  - 8.6.4. Interazione basata su API
- 8.7. Sviluppo *cloud-native*
  - 8.7.1. Sviluppo nativo *cloud*
  - 8.7.2. Contenitori e piattaforme di orchestrazione dei contenitori
  - 8.7.3. Integrazione continua nel cloud
  - 8.7.4. Salvataggio degli eventi nel cloud
- 8.8. Infrastruttura come codice nel cloud
  - 8.8.1. Automazione della gestione e del provisioning nel cloud
  - 8.8.2. Terraform
  - 8.8.3. Integrazione con *scripting*
- 8.9. Creare un'infrastruttura ibrida
  - 8.9.1. Interconnessione
  - 8.9.2. Interconnessione con *datacenter*
  - 8.9.3. Interconnessione con altri cloud
- 8.10. Computing con elevate prestazioni
  - 8.10.1. Computing con elevate prestazioni
  - 8.10.2. Creazione di un Cluster con prestazioni elevate
  - 8.10.3. Applicazione del computing con elevate prestazioni

## Modulo 9. Modelli e semantica formale. Programmazione orientata al Calcolo Distribuito

- 9.1. Modello Semantico di dati
  - 9.1.1. Modello Semantico di dati
  - 9.1.2. Modello Semantico di dati. Propositi
  - 9.1.3. Modello Semantico di dati. Applicazioni
- 9.2. Modello semantico dei linguaggi di programmazione
  - 9.2.1. Processori linguistici
  - 9.2.2. Traduzione e interpretazione
  - 9.2.3. Linguaggi ibridi
- 9.3. Modelli di computer
  - 9.3.1. Calcolo monolitico
  - 9.3.2. Calcolo parallelo
  - 9.3.3. Calcolo distribuito
  - 9.3.4. Calcolo cooperativo (P2P)
- 9.4. Calcolo parallelo
  - 9.4.1. Architetture parallele
  - 9.4.2. Hardware
  - 9.4.3. Software
- 9.5. Modelli distribuiti. *Grid Computing*
  - 9.5.1. Architettura *Grid Computing*
  - 9.5.2. Architettura *Grid Computing*. Analisi
  - 9.5.3. Architettura *Grid Computing*. Applicazioni
- 9.6. Modelli distribuiti. *Cluster Computing*
  - 9.6.1. Architettura *Cluster Computing*
  - 9.6.2. Architettura *Cluster Computing*. Analisi
  - 9.6.3. Architettura *Cluster Computing*. Applicazioni
- 9.7. *Cluster Computing*. Strumenti attuali per implementarlo. Ipersori
  - 9.7.1. Concorrenti di mercato
  - 9.7.2. VMware Hypervisor
  - 9.7.3. Hyper-V

- 9.8. Modelli distribuiti. *Cloud Computing*
  - 9.8.1. Architettura *Cloud Computing*
  - 9.8.2. Architettura *Cloud Computing*. Analisi
  - 9.8.3. Architettura *Cloud Computing*. Applicazioni
- 9.9. Modelli distribuiti. *Cloud Computing Amazon*
  - 9.9.1. *Cloud Computing Amazon*. Funzionalità
  - 9.9.2. *Cloud Computing Amazon*. Licenze
  - 9.9.3. *Cloud Computing Amazon*. Architettura di riferimento
- 9.10. Modelli distribuiti. *Cloud computing Microsoft*
  - 9.10.1. *Cloud computing Microsoft*. Funzionalità
  - 9.10.2. *Cloud computing Microsoft*. Licenze
  - 9.10.3. *Cloud computing Microsoft*. Architettura di riferimento

## Modulo 10. Applicazioni del Calcolo Parallelo e Distribuito

- 10.1. Calcolo parallelo e distribuito nelle applicazioni attuali
  - 10.1.1. Hardware
  - 10.1.2. Software
  - 10.1.3. Importanza dei tempi
- 10.2. Clima. Cambiamento climatico
  - 10.2.1. Applicazioni al clima. Fonti di dati
  - 10.2.2. Applicazioni al clima. Quantità di dati
  - 10.2.3. Applicazioni al clima. In tempo reale
- 10.3. GPU Calcolo Parallelo
  - 10.3.1. GPU Calcolo Parallelo
  - 10.3.2. GPU vs. CPU. Uso di GPU
  - 10.3.3. GPU. Esempi
- 10.4. Smart *Grid*. Calcolo nelle reti elettriche
  - 10.4.1. Smart *Grid*
  - 10.4.2. Modelli concettuali. Esempi
  - 10.4.3. Smart *Grid*. Esempio
- 10.5. Motore distribuito. *ElasticSearch*
  - 10.5.1. Motore distribuito. *ElasticSearch*
  - 10.5.2. Architettura con *Elasticsearch*. Esempi
  - 10.5.3. Motore distribuito. Casi d'uso
- 10.6. *Big Data Framework*
  - 10.6.1. *Big Data Framework*
  - 10.6.2. Architettura degli strumenti avanzati
  - 10.6.3. *Big Data* nel Calcolo Distribuito
- 10.7. Database in memoria
  - 10.7.1. Database in memoria
  - 10.7.2. Soluzione di Redis. Casi di successo
  - 10.7.3. Implementazione di soluzioni con database in memoria
- 10.8. *Blockchain*
  - 10.8.1. Architettura *Blockchain*. Componenti
  - 10.8.2. Collaborazione tra nodi e consensi
  - 10.8.3. Soluzioni *Blockchain*. Implementazione
- 10.9. Sistemi Distribuiti in Medicina
  - 10.9.1. Componenti dell'architettura
  - 10.9.2. Sistemi Distribuiti in Medicina. Funzionamento
  - 10.9.3. Sistemi Distribuiti in Medicina. Applicazioni
- 10.10. Sistemi distribuiti nel settore aereo
  - 10.10.1. Progettazione dell'architettura
  - 10.10.2. Sistemi distribuiti nel settore aereo. Funzionalità dei componenti
  - 10.10.3. Sistemi distribuiti nel settore aereo. Applicazioni



Questo programma è la spinta necessaria per ottenere il miglioramento professionale che stavi cercando e per il quale hai lavorato sodo"

06

# 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***.



“

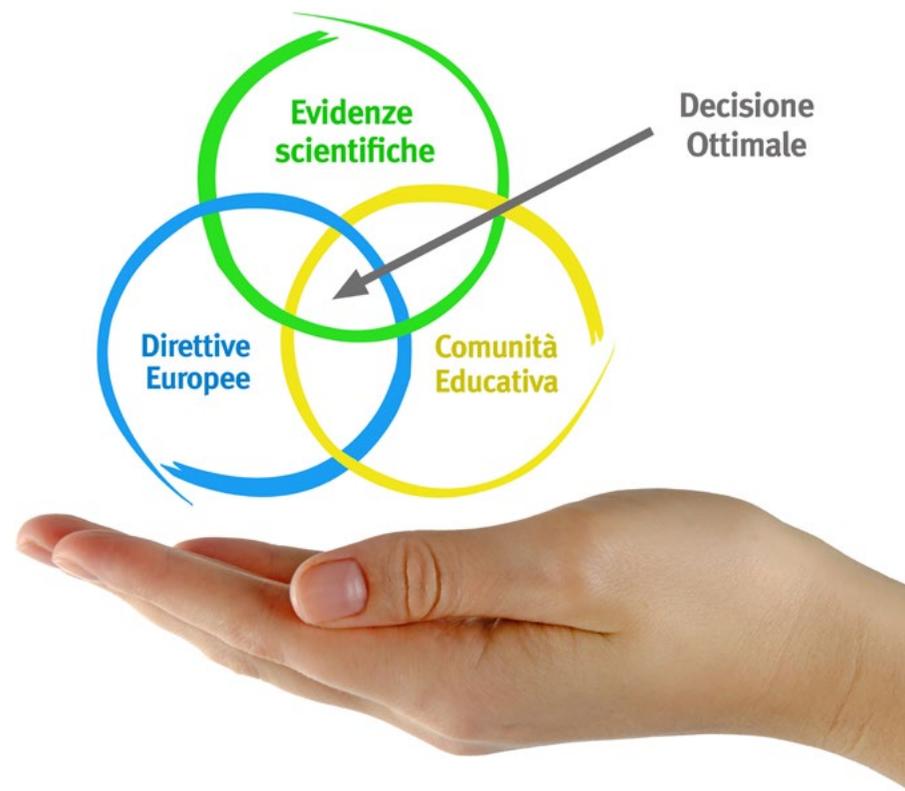
*Scopri il Relearning, un sistema che abbandona l'apprendimento lineare convenzionale, per guidarti attraverso dei sistemi di insegnamento ciclici: una modalità di apprendimento che ha dimostrato la sua enorme efficacia, soprattutto nelle materie che richiedono la memorizzazione”*

## 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 Scuole di Informatica del mondo da quando esistono. 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 corso, 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 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 imparerai con 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.



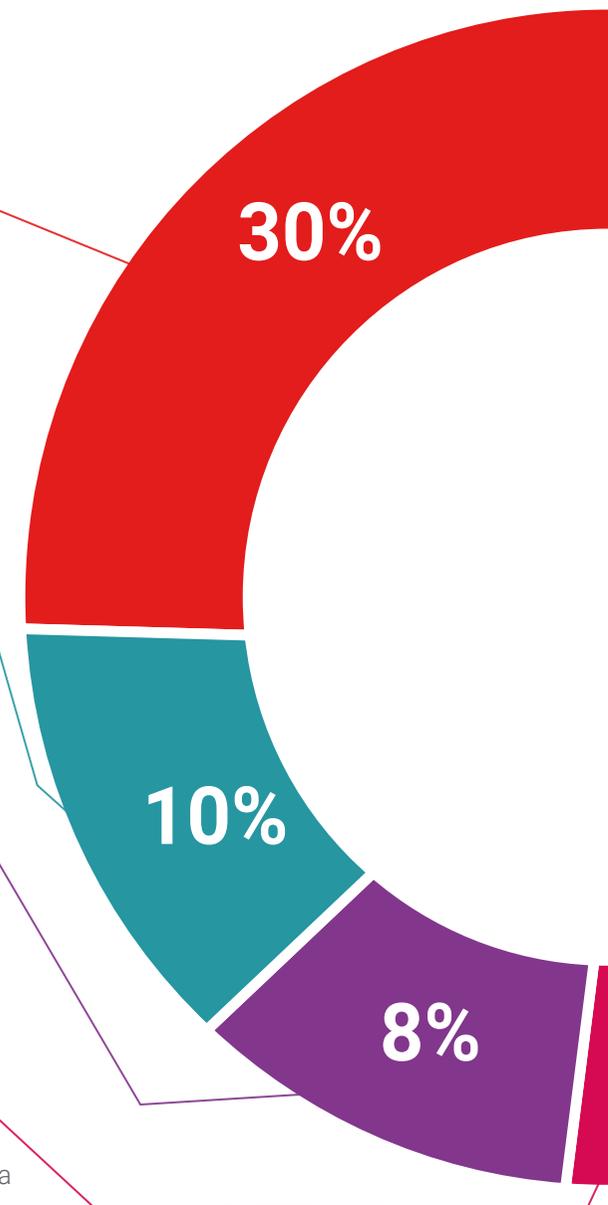
#### Pratiche di competenze e competenze

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.



# 07 Titolo

Il Master Privato in Calcolo Parallelo e Distribuito 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”*

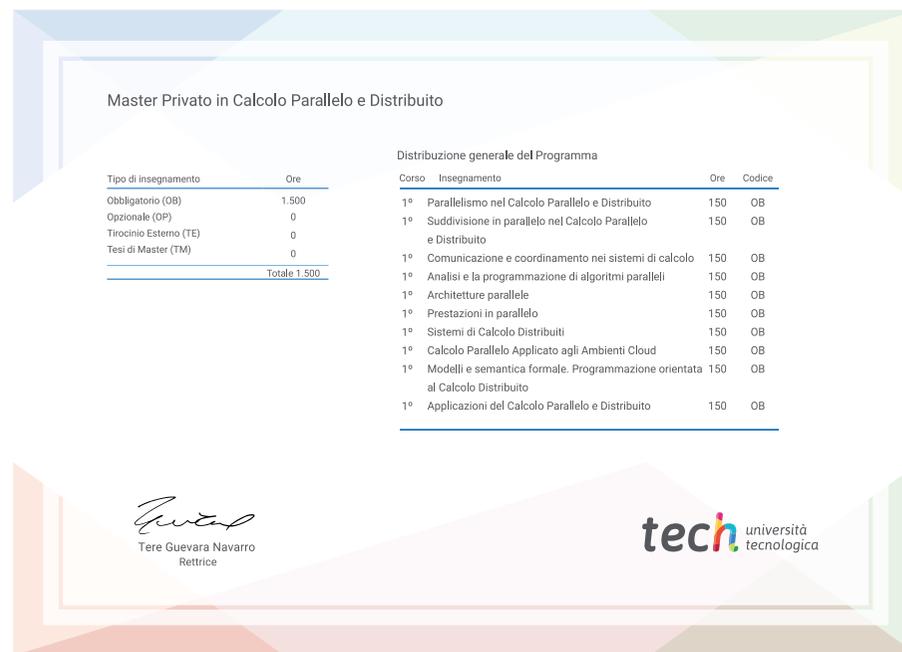
Questo **Master Privato in Calcolo Parallelo e Distribuito** possiede il programma più completo e aggiornato del mercato.

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 Calcolo Parallelo e Distribuito**

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 lingue

**tech** università  
tecnologica

**Master Privato**  
Calcolo Parallelo  
e Distribuito

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

# Master Privato

## Calcolo Parallelo e Distribuito

