

# Architetture dei Sistemi Distribuiti [ 2205104 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** LUCA VOLLERO

**Periodo:** Secondo Ciclo Semestrale

## Obiettivi formativi

L'insegnamento impartirà allo studente conoscenze e competenze relative alle architetture hardware e software dei sistemi informatici con particolare riferimento ai sistemi in grado di acquisire, memorizzare ed elaborare, offline e online, dati provenienti da sensori e da altre sorgenti di dati distribuite nell'ambiente fisico, con il coinvolgimento di sistemi intelligenti per il processamento intermedio dei dati e il tuning dei sistemi coinvolti. Lo studente è inoltre introdotto ai metodi e alle tecniche per la configurazione e il dimensionamento di tali sistemi allo scopo di ottimizzarne le prestazioni.

## Prerequisiti

-

## Contenuti del corso

Sistemi Distribuiti: Introduzione e tassonomia

Architettura dei Sistemi:

- Modelli centralizzati
- Modelli distribuiti
- Modelli ibridi
- Middleware

Processi:

- Fondamenti di sistemi operativi
- Processi e thread
- Comunicazione tra processi e thread
  - Il paradigma Client&Server
  - Il paradigma Publish-Subscribe
- Sistemi virtuali

Fondamenti di rete:

- Rete di computer
- Il modello RPC
- Il modello a scambio di messaggi
- Comunicazione basata su stream
- Multicast

Sincronizzazione:

- Sincronizzazione degli orologi
- Orologi logici
- Accesso sincronizzato nei sistemi distribuiti
- Sistemi di polling

IoT e Sistemi Distribuiti

- Applicazioni IoT
- Servizi web
- Cloud computing e Platform as a Service
- Edge computing
- Fog Computing

Integrità e replicazione dei dati:

- Modelli data-centric
- Modelli client-centric
- Gestione delle repliche
- Protocolli di integrità dei dati

Sistemi Fault Tolerant

Sicurezza dei Sistemi Distribuiti

### **Metodi didattici**

Il corso consiste in lezioni teoriche frontali (48 ore), esercitazioni in laboratorio (12 ore) e lo sviluppo completo di un progetto finalizzato ad applicare le conoscenze e competenze acquisite (12 ore).

### **Modalità di verifica dell'apprendimento**

L'esame ha lo scopo di verificare l'acquisizione delle conoscenze e delle abilità specificate negli obiettivi formativi dell'insegnamento. L'esame consiste nella discussione di un progetto sviluppato dallo studente e un colloquio orale sugli argomenti di teoria oggetto del programma del corso. Nella discussione del progetto lo studente deve dimostrare di conoscere e aver saputo applicare i modelli e le metodologie dei sistemi distribuiti, attraverso la costruzione di un'analisi dei requisiti puntuale e l'applicazione delle conoscenze sulle architetture e sulle tecnologie oggetto del corso. Nell'esame orale lo studente sarà chiamato a dimostrare le sue conoscenze e competenze nella soluzione di problemi specifici relativi sempre la progettazione, descrizione e gestione di sistemi distribuiti.

Il voto finale si compone per metà della valutazione della qualità tecnica dell'elaborato sviluppato (soluzione architeturale e documentazione) unita alla padronanza con cui verrà condotta la discussione relativa al suo sviluppo, e per l'altra metà dalla dimostrazione della conoscenza degli argomenti del corso unita alla chiarezza con cui questi verranno esposti.

L'attribuzione della lode è condizionata da una valutazione eccellente sui punti precedentemente elencati e la dimostrazione da parte del discente di riuscire a gestire in modo maturo e profondo problematiche di progettazione e gestione non trattate in modo diretto durante il corso.

### **Testi di riferimento**

- Andrew S. Tanenbaum, Maarten Van Steen, Distributed Systems: Principles and Paradigms, Pearson
- David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press.
- Dispense del docente (su piattaforma e-learning)

### **Altre informazioni**

Conoscenza e capacità di comprensione.

Il corso trasferirà allo studente le seguenti conoscenze e capacità di comprensione:

- Conoscenza e comprensione delle diverse architetture distribuite in generale e di quelle IoT in particolare, dei loro componenti funzionali e delle loro caratteristiche.
- Conoscenza delle tecnologie hardware e software di interconnessione dei dispositivi e comprensione delle loro caratteristiche prestazionali.
- Conoscenza e comprensione delle soluzioni architetture e funzionali per applicazioni software distribuite, in generale, e IoT, in particolare.
- Conoscenza e comprensione dei metodi di descrizione e progettazione di sistemi distribuiti, in generale, e IoT, in particolare, e delle metodiche di configurare e dimensionare dei loro componenti.

Capacità di applicare conoscenza e comprensione.

Al termine del corso lo studente sarà in grado di:

- Analizzare e definire le specifiche dei requisiti di un sistema distribuito/IoT per l'acquisizione, la memorizzazione e l'elaborazione di dati.
- Selezionare soluzioni architetture e funzionali per applicazioni distribuite/IoT di acquisizione, memorizzazione ed elaborazione di dati.
- Configurare e dimensionare l'infrastruttura di un sistema distribuit/IoT per l'acquisizione, la memorizzazione e l'elaborazione di dati.
- Selezionare le tecnologie di un sistema distribuito per l'acquisizione, la memorizzazione e l'elaborazione di dati.

Abilità comunicative.

Lo studente svilupperà la capacità di descrivere e comunicare, verbalmente e per iscritto, in modo formale, rigoroso e preciso l'architettura funzionale e le interazioni tra i componenti di un sistema distribuito/IoT.

Capacità di apprendere.

Lo studente sarà in grado di acquisire nuove conoscenze e competenze relativamente a tecnologie e soluzioni architetture di sistemi distribuiti/IoT.

**L'attività didattica è offerta in:**

**Facoltà Dipartimentale di Ingegneria**

<b>Tipo corso</b>	<b>Corso di studio (Ordinamento)</b>	<b>Percorso</b>	<b>Crediti</b>	<b>S.S.D.</b>
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	9	ING-INF/05

*Stampa del 06/11/2025*

# Digital Twins per il Controllo, l'Automazione e la Predictive Maintenance [ 2205102 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** ROBERTO SETOLA

**Periodo:** Primo Ciclo Semestrale

## Obiettivi formativi

Il corso mira a fornire conoscenze teoriche e pratiche sull'utilizzo di tecniche di controllo moderno basate sul concetto di stato e di come questo possa essere dedotto attraverso l'utilizzo di un Digital Twin. Gli studenti apprenderanno l'uso di metodologie avanzate per la modellazione e la simulazione di sistemi reali, integrando controllo, sistemi di osservazione come il filtro di Kalman, tecniche di rilevamento di guasti. Il corso prevede inoltre l'utilizzo di algoritmi di Digital Twin all'interno di schemi di automazione industriale.

## Prerequisiti

Elementi basilari di Teoria del Controllo (Fondamenti di Automatica)

## Contenuti del corso

Concetto di Stato e Tecniche Controllo Moderno (25%): Richiami sul concetto di stato e relative proprietà dei sistemi lineari. Sistemi a tempo discreto e sistemi a dati campionati. Tecniche di controllo a retroazione di stato e tecniche di controllo ottimo. Introduzione ai concetti di controllo ottimo e applicazione ai Digital Twins per sistemi dinamici.

Digital Twins (25%): Introduzione ai sistemi di Digital Twin. Tecniche di progettazione di sistemi di Digital Twin basate su approcci model-driven: Osservatore di Luenberger, Filtro di Kalman e Filtro di Kalman Esteso. Elementi per la realizzazione di Digital Twin mediante tecniche data-driven. Utilizzo di schemi di Digital Twins per la stima e la predizione dello stato di sistemi complessi e per la chiusura dei loop di controllo.

Tecniche di Anomaly e Fault Detection (25%): Approcci per il rilevamento di situazioni anomale e di fault. Metodologie per la progettazione di algoritmi in grado di identificare situazione anomale (Anomaly Detection System) e guasti (Fault Detection System) e per la loro prevenzione (Predictive Maintenance) e gestione (Incident Response System).

Algoritmi Avanzati (25%): Osservatori Unknown Input Observer per la stima di un disturbo ignoto. Osservatori di sistemi soggetti a rumori non Gaussiani tramite approcci di tipo Zonotopico. Approcci Data-Driven (es. Stima ai minimi quadrati, stima a massima verosimiglianza, variational Autoencoders).

## Metodi didattici

Lezioni frontali (60%). Esercitazioni alla lavagna, con il calcolatore e/o sperimentali (40%).

## Modalità di verifica dell'apprendimento

Il voto è composto per il 40% dalla valutazione del progetto e per l'60% dalla valutazione del colloquio orale. Il colloquio orale si struttura in due-tre domande teoriche (es., dimostrazioni o presentazione di schemi di controllo) o pratiche (es. esercizi).

## Testi di riferimento

Dispense fornite dal docente

## Altre informazioni

Il corso mira a fornire le seguenti competenze:

- 1) Conoscenza e capacità di comprensione:
  - Conoscenza e strumenti metodologici per la progettazione di algoritmi di controllo moderni basati sull'utilizzo dello stato di un sistema.
  - Conoscenza di strumenti metodologici per la realizzazione di sistemi di Digital Twin per la determinazione dello stato di un sistema.
  - Comprensione dell'utilizzo degli algoritmi di stima come il filtro di Kalman all'interno dei modelli di Digital Twins.
  - Conoscenza delle principali tecniche per la individuazione automatica di situazioni anomale e di guasto (fault)

detection) e per la loro gestione predittiva e correttiva (predictive maintenance e fault management) utilizzando Digital Twins.

- Capacità di integrare algoritmi di supervisione, gestione e controllo basate su concetto del Digital Twin all'interno di schemi di controllo di automazione industriale.

- Capacità di utilizzo di algoritmi avanzati come metodi data-driven, Zonotopes, e/o Unknown Input Observers

2) Conoscenza e capacità di comprensione applicate:

- Progettazione di algoritmi di controllo e di monitoraggio basati sul concetto di stato di un sistema e sua ricostruzione attraverso l'impiego di un Digital Twins, considerando le problematiche legate all'implementazione su calcolatori o microcontrollori.

3) Autonomia di giudizio:

- Capacità di selezionare la strategia più efficace per l'analisi e il controllo di sistemi complessi attraverso l'uso di Digital Twins.

4) Abilità comunicative:

- Capacità di discutere e collaborare con docenti e colleghi su argomenti avanzati legati al controllo moderno, all'utilizzo dei Digital Twins, alla manutenzione predittiva e all'automazione industriale.

- Abilità nel comunicare soluzioni tecniche e nel formulare domande e risposte pertinenti in lingua inglese.

5) Capacità di apprendere:

- Metodo di apprendimento basato sulla partecipazione attiva agli insegnamenti teorici e alle esercitazioni pratiche.

- Stimolazione della capacità di risolvere problemi pratici attraverso l'uso di Digital Twins, con un incremento progressivo della difficoltà durante il corso.

**L'attività didattica è offerta in:**

**Facoltà Dipartimentale di Ingegneria**

<b>Tipo corso</b>	<b>Corso di studio (Ordinamento)</b>	<b>Percorso</b>	<b>Crediti</b>	<b>S.S.D.</b>
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	9	ING-INF/04

*Stampa del 06/11/2025*

# Fondamenti di Cybersecurity [ 2205105 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** LUCA FARAMONDI

**Periodo:** Secondo Ciclo Semestrale

## Obiettivi formativi

Il corso si prefigge di fornire al discente i principali concetti di cybersecurity al fine di consentirgli di introdurre tali elementi in fase di progetto, gestione, manutenzione e dismissione di un sistema cyber-physical valutando i dettagli architetturali sia infrastrutturali che applicativi.

Inoltre, il corso si propone di fornire le conoscenze di base circa gli standard di sicurezza di dati fornendo un insieme di best practice necessarie per la gestione della sicurezza delle informazioni.

## Prerequisiti

Basi di programmazione.

## Contenuti del corso

Parte I – Introduzione alle Operational Technologies

Richiami di automazione

Sistemi di controllo (richiami). Sensori ed attuatori (richiami). Sistemi SCADA. Controllori industriali.

Mezzi trasmissivi.

Operational Technology ed Information Technology

Architetture ed elementi caratteristici degli impianti. Gestione degli aggiornamenti. Dispositivi IoT ed IIoT.

Sicurezza dei processi e caratteristiche di integrità, disponibilità e confidenzialità dei dati.

Parte II – La Sicurezza delle reti informatiche

Architetture di rete

Dispositivi di rete. Topologie di rete. Tecniche di commutazione. Gestione della comunicazione.

Protocolli di comunicazione

Protocolli della suite TCP-IP. Protocollo ModBUS. Standard OPC UA. Tecniche di Crittografia. Protocolli di scambio email, SMTP e phishing.

Vulnerabilità delle reti e dei protocolli

Vulnerabilità dei protocolli industriali. Vulnerabilità delle reti wireless. Tecniche di attacco (ARP Spoofing, Man in the Middle, Denial of Service, Distributed Denial of Service).

Strumenti per Penetration Testing e Vulnerability Assessment

La distribuzione Kali Linux. Strumenti per simulazioni hardware in the loop. Mini CPS.

Soluzioni per il monitoraggio della rete

Intrusion detection Systems, Intrusion. Prevention Systems. Firewall. DMZ. Snort. Soluzioni Anomaly Based e Signature Based per l'identificazione delle minacce.

Parte III – La Sicurezza dei sistemi IT

Sistemi di autenticazione ed accesso a servizi

Caratteristiche della sicurezza dei dati. Certificati digitali. Firma digitale. Servizi di posta. Metodi di autenticazione.

Sicurezza di dispositivi IoT

Dispositivi esposti in rete, configurazioni di rete per la condivisione sicura di periferiche: webcam, stampanti, etc.

Parte IV – La sicurezza del software

Vulnerabilità e minacce software. Security by Design. Approcci Design by Contract e Defensive programming. Code Injection. SQL Injection. Buffer Overflow. Privilege Escalation. Cross-Site Scripting, Vulnerabilità su piattaforme web.

Parte V – Analisi delle Fonti Aperte

Social Engineering. Strumenti per indagini OSINT. Shodan.IO

## Metodi didattici

Lezioni frontali sugli argomenti del corso (50%)

Esercitazioni in aula ed in laboratorio (30%)

Seminari con esperti (20%)

### Modalità di verifica dell'apprendimento

Sia per la prova scritta che l'elaborato progettuale verrà attribuito un punteggio espresso in trentesimi in relazione alla correttezza, esaustività e complessità del lavoro presentato. Una valutazione positiva sia della prova scritta che del progetto (pari, almeno, a 18/30) è condizione necessaria per il superamento dell'esame. Il voto finale sarà calcolato come media ponderata del voto della prova scritta (40%) e della valutazione del progetto di gruppo (60%).

### Testi di riferimento

Materiale didattico di supporto all'apprendimento

Dispense del docente

Articoli Scientifici

Testi di Consultazione

James F. Kurose, Keith W. Ross "Internet e Reti di Calcolatori" Pearson Education.

C. P. Pfleeger, S. L. Pfleeger, J. Margulies: "Security in Computing, 5th Edition", Prentice Hall, 2015

Charles J. Brooks, Christopher Grow, Philip Craig, Donald Short, "Cybersecurity Essentials", Sybex Inc, 2018

Alan Calder, Steve Watkins "IT Governance: An International Guide to Data Security and ISO 27001/ISO 27002", Kogan Page Ltd, 2019

### Altre informazioni

- Consapevolezza dei legami tra processi fisici e infrastruttura informatica
- Conoscenza dei principali protocolli di comunicazione nelle reti e loro vulnerabilità
- Capacità di progettare infrastrutture di rete e software per applicazioni cyber-physical minimizzando il rischio cyber.
- Capacità di organizzare attività di penetration testing e vulnerability assessment sistemi cyber-physical
- Capacità di progettare soluzioni per l'identificazione e la mitigazione di minacce cyber
- Capacità di valutare le minacce legate a sistemi IT e servizi web.
- Conoscenza delle principali best practice che regolano la gestione della sicurezza delle informazioni

**L'attività didattica è offerta in:**

### Facoltà Dipartimentale di Ingegneria

Tipo corso	Corso di studio (Ordinamento)	Percorso	Crediti	S.S.D.
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	9	ING-INF/04

*Stampa del 06/11/2025*

# Fondamenti di Intelligenza Artificiale: strumenti di sviluppo e metodi [ 2205103 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** PAOLO SODA

**Periodo:** Ciclo Annuale Unico

## Obiettivi formativi

Il corso si compone di due moduli: Strumenti di Sviluppo e Metodi.

Gli obiettivi di apprendimento del modulo Strumenti di Sviluppo forniscono allo studente conoscenze e competenze necessarie per utilizzare linguaggi di programmazione di alto livello per lo sviluppo di applicazioni software orientate all'elaborazione dei dati. Lo studente approfondirà l'uso di metodi e strumenti di programmazione che consentono lo sviluppo efficiente del software attraverso la generazione e il riutilizzo di componenti modulari di alta qualità. Le competenze di programmazione sono applicate a strategie e algoritmi per l'analisi dei dati nelle applicazioni di Intelligenza Artificiale.

Gli obiettivi di apprendimento del modulo Metodi sono acquisire i concetti di base del Machine Learning (ML) e dell'IA simbolica, ovvero i sistemi e gli algoritmi che si basano sull'osservazione dei dati per la sintesi di nuova conoscenza. Ad esempio, l'apprendimento può avvenire catturando caratteristiche da esempi, strutture dati o sensori, per analizzare e valutare le relazioni tra le variabili osservate. In particolare, lo studente dovrebbe:

- Acquisire un adeguato livello di conoscenza dei fondamenti teorici dei principali modelli computazionali per l'apprendimento (ad es. apprendimento supervisionato e non supervisionato, classificatori e regressori, modelli di apprendimento basati su distanza e su modello, classificatori lineari e a kernel, modelli evolutivi, analisi delle serie temporali, ecc.);
- Comprendere i metodi per la sintesi di nuova conoscenza;
- Comprendere i fondamenti dei metodi per definire una procedura sperimentale e per la valutazione delle prestazioni;
- Comprendere il potenziale dell'IA per lo sviluppo di sistemi di supporto alle decisioni, data mining e analisi dei big data;
- Imparare l'uso di ambienti di sviluppo appropriati per l'applicazione dei metodi IA.

## Prerequisiti

Le conoscenze e le competenze richieste per l'ammissione al corso di laurea sono prerequisiti. Si raccomanda inoltre di superare l'esame finale del corso di Ottimizzazione.

## Contenuti del corso

Modulo Development Tools

- Elementi di base del linguaggio Python (4 ore, tenute dal titolare del corso)
- Built-in types
- Variabili
- Oggetti e metodi
- Le strutture dati in Python: liste, tuple, set, dizionari
- Strutture di controllo
- Funzioni e passaggio dei parametri
- Ricorsione
- Pacchetti
- Rappresentazione dei dati (6 ore, tenute dal titolare del corso)
- Il concetto di informazione e riferimenti alla rappresentazione dei dati
- Strutture dati: modelli e implementazione
- Riferimenti al modello relazionale
- Il modello associativo e oggetto
- Rappresentazione serializzata dei dati: JSON, YAML e XML
- Programmazione orientata agli oggetti (10 ore, tenute dal titolare del corso e dal co-docente)
- Il concetto di classe, sottoclasse e interfaccia
- Modularity e decoupling
- Information hiding
- Ereditarietà e Polimorfismo
- Design Patterns
- Programmazione orientata agli oggetti in Python



- Integrazioni di programmazione Python (4 ore, tenute dal titolare del corso)
- Programmazione funzionale in Python
- Espressioni lambda
- Le funzioni map e filter
- List comprehension
- Pacchetti rilevanti (6 ore, tenute dal co-docente)
- numpy
- scipy
- pandas
- Sviluppo e organizzazione del software (10 ore, tenute dal co-docente)
- Modelli di sviluppo software collaborativo
- Strumenti di modellazione dei sistemi software: UML (Diagramma delle classi, Diagramma dei casi d'uso, Diagramma di sequenza)
- Strumenti per il versionamento del codice sorgente (git, GitHub)
- 
- Test Driven Development (TDD)
- Sviluppo software basato su containers
- Il paradigma Infrastructure-as-Code (IaC)
- Sviluppo di un progetto (20 ore, tenute dal titolare del corso, dai co-docenti)

#### Modulo Metodi

- Introduzione, definizione del concetto di apprendimento e riconoscimento dei modelli, varie definizioni, metodologia e processo di analisi, concetto di descrittore o caratteristica
- Apprendimento supervisionato, non supervisionato, semi-supervisionato, reinforcement learning
- Modello del processo di analisi dei dati, esempio semplice di cosa significhi classificare basandosi sull'esperienza, esempio di riconoscimento dei pesci, overfitting, underfitting
- Metodi di validazione
- Metodi di valutazione delle prestazioni, metodologie sperimentali, cross-validation, matrice di confusione, metriche derivate dalla matrice, confusione, la curva ROC
- Teorema di Bayes, teoria decisionale bayesiana e il classificatore bayesiano
- Classificatore non parametrico: Nearest Neighbor (NN) e la sua estensione (kNN), affidabilità delle decisioni di kNN, considerazioni computazionali
- Support Vector Machine (SVM): algoritmo di apprendimento, tipi di kernel, il problema dell'affidabilità delle decisioni XOR
- Alberi decisionali (CART, ID3, C4.5); l'albero come strumento per la regressione e la selezione delle caratteristiche
- Metodi di classificazione basati sulla decomposizione binaria dei problemi multiclasse
- Sistemi multi-esperto: bagging e boosting, adaboost; random forest
- Riduzione della dimensionalità: Feature selection ed Feature extraction
- Preparazione e pulizia dei dati: Pulizia dei dati, Dati mancanti, Dati errati e incoerenti, Scaling e normalizzazione, Riduzione e trasformazione dei dati, Campionamento dei dati, Riduzione dei dati con trasformazione del tipo (per le serie temporali)
- Introduzione alle reti neurali e al deep learning. Modello di neurone, funzioni di trasferimento, il perceptrone, LMS, discesa gradiente stocastica, introduzione a MLP, il problema della saturazione, Error-Backpropagation, cross-entropy, Softmax, ReLU, Tecniche per combattere l'overfitting (ad es. regolarizzazione L1 e L2, dropout), il problema del gradiente evanescente o esplosivo
- Apprendimento non supervisionato, clustering e stima delle prestazioni: definizione di apprendimento non supervisionato, nomenclatura e principali applicazioni, conoscenza teorica e algoritmica dei principali algoritmi di clustering (Agglomerativo Gerarchico, K-means, DB-SCAN, Expectation Maximization, Mean Shift e Clustering Spettrale), conoscenza delle principali metriche per la stima delle prestazioni interne ed esterne (indice di Silhouette e indice di Rand corretto)
- Metodi di regressione (lineare, logistica, kNN, alberi, SVR) e stima delle prestazioni;
- Introduzione al reinforcement learning;
- Introduzione all'analisi delle serie temporali: definizioni di serie temporali univariate e multivariate, riduzione del rumore di preprocessing, rappresentazione (Trasformata di Fourier Discreta e Approssimazione Aggregata Simbolica), previsione con modelli autoregressivi, classificazione con Dynamic Time Warping e Time-Delay Neural Networks, suggerimenti sulla clusterizzazione delle serie temporali e tassonomia.
- AutoEncoders: undercomplete e overcomplete
- Modelli evolutivi: evoluzione naturale e genetica molecolare, sistemi evolutivi artificiali, Intelligenza di Sciame;
- Attività di laboratorio: utilizzo delle librerie Python per l'IA, anche in cloud

#### Metodi didattici

Il corso consiste in lezioni teoriche frontali (55%), lezioni capovolte (10%), esercitazioni di laboratorio con l'uso di strumenti open source (20%) e lo sviluppo completo di uno o più progetti in piccoli gruppi (15% più lavoro individuale) volti ad applicare le conoscenze e le competenze acquisite.

#### Modalità di verifica dell'apprendimento

Metodi di valutazione della conoscenza e criteri:

I risultati specifici di apprendimento vengono verificati attraverso lo sviluppo di due progetti realizzati in piccoli gruppi e sotto la supervisione di uno dei docenti e un orale.

Lo sviluppo del primo progetto è costantemente monitorato dal docente attraverso la piattaforma di sviluppo collaborativo introdotta durante il corso, verificando il contributo di ogni membro del gruppo, i metodi di interazione tra i membri e la qualità del lavoro svolto dal gruppo nel suo insieme. La valutazione delle conoscenze e delle competenze acquisite da ciascuno studente è completata durante un esame finale consistente nella discussione del progetto e degli argomenti teorici inclusi nel programma.

Il secondo consiste in un lavoro sperimentale da presentare in aula o in un colloquio orale. Lo scopo di questo test è verificare che lo studente abbia acquisito la capacità di utilizzare modelli computazionali per risolvere problemi di classificazione, clustering e regressione, attraverso l'uso di strumenti software disponibili per l'applicazione dei metodi ML.

Agli studenti verrà fornito un dataset reale che specifica il problema da risolvere; ad esempio, può essere fornito un dataset con segnali acquisiti da un dispositivo IoT, richiedendo agli studenti di sviluppare un algoritmo in grado di prevedere il valore futuro del segnale stesso.

Nel lavoro sperimentale, gli elementi presi in considerazione sono: la logica seguita dallo studente nella risoluzione del problema, la correttezza della procedura identificata per la soluzione, l'adeguatezza della soluzione proposta in relazione alle competenze che lo studente si aspetta di aver acquisito alla fine del corso. Ciascuno di questi elementi pesa ugualmente nella valutazione della prova di laboratorio, e il soddisfacimento di questi aspetti, almeno al 60%, è una condizione necessaria per ottenere un voto di 18. Voti più alti verranno assegnati agli studenti i cui lavori soddisfano tutti gli aspetti sopra elencati, in proporzione crescente.

Il test orale consiste in un colloquio, che mira a verificare che lo studente abbia acquisito un livello adeguato di conoscenza dei fondamenti teorici dei principali modelli computazionali per l'IA. Durante il test orale, gli elementi presi in considerazione sono: la logica seguita dallo studente nel formulare la risposta alla domanda, la correttezza della procedura identificata per la soluzione della domanda, l'adeguatezza della soluzione proposta in relazione alle competenze che si suppone lo studente abbia acquisito alla fine del corso, l'uso di un linguaggio appropriato.

Ciascuno di questi elementi pesa ugualmente nella valutazione del test orale, e il soddisfacimento di questi aspetti, almeno al 60%, è una condizione necessaria per ottenere un voto di 18. Voti più alti verranno assegnati agli studenti i cui lavori soddisfano tutti gli aspetti elencati sopra, in proporzione crescente.

Un esempio di domanda potrebbe essere: "presenta il modello degli alberi decisionali".

Criteri per la misurazione dell'apprendimento e la definizione del voto finale:

La valutazione è su trenta con la possibile attribuzione della lode.

Il voto finale dipende dal primo progetto (30%) il secondo progetto (20%) e dal test orale (50%).

La valutazione dei due progetti dipende dalla valutazione della qualità tecnica del lavoro sviluppato (documentazione esterna e interna e codifica del software) e dall'attività collaborativa svolta, combinata con la maestria con cui viene condotta la discussione relativa al suo sviluppo.

Rispetto al secondo progetto, il gruppo che ottiene i migliori risultati nel lavoro sperimentale, in ogni caso al di sopra di una certa soglia minima determinata in base alla complessità del problema, avrà diritto a +1/+2 punti sul voto finale.

Per ciascuno dei tre componenti, il voto finale è tra 18 e 23 se lo studente dimostra di aver raggiunto un livello appena sufficiente, tra 24 e 27 se lo studente dimostra di aver raggiunto un livello adeguato, tra 28 e 30 se lo studente dimostra di aver raggiunto un alto livello, ovvero lo studente deve dimostrare di aver acquisito una conoscenza eccellente di tutti gli argomenti trattati nel corso, essendo in grado di collegarli in modo logico e coerente. La lode si ottiene dimostrando un alto grado di conoscenza degli argomenti e degli strumenti di simulazione, dimostrando un alto grado di autonomia e giudizio, e mostrando un'alta qualità di esposizione.

### Testi di riferimento

- Luciano Ramalho, *Fluent Python*, O'Reilly
- Ferdinando Santacroce, *Git Essentials*, Packt (<https://www.packtpub.com/product/git-essentials/9781785287909>)
- Documentazione dei pacchetti Python
- Bishop, *Pattern recognition and machine learning*. Springer, 2006
- Duda, et al. *Pattern classification*. John Wiley & Sons, 2012
- Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani - *An Introduction to Statistical Learning with Applications in R*. Springer Texts in Statistics
- Aurélien Géron - *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Techniques and Tools to Build Learning Machines*, O'Reilly;
- Charu C. Aggarwal - *Data Mining The Textbook*. Springer 2015.
- Charu C. Aggarwal, Chandan K. Reddy - *Data Clustering: Algorithms and Applications*
- Dario Floreano, Claudio Matusi - *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*
- Appunti forniti dal docente

### Altre informazioni

Conoscenza e comprensione.

Il corso trasferirà allo studente le seguenti conoscenze e comprensioni:

- Conoscenza e comprensione delle tecniche di analisi dei requisiti finalizzate allo sviluppo di componenti software che soddisfino tali requisiti.
- Conoscenza di uno o più linguaggi di programmazione che supportano lo sviluppo modulare e il riutilizzo del

software in un ambiente distribuito.

- Conoscenza delle metodologie per la documentazione del software e per la verifica della qualità del software.
- Conoscenza e comprensione della produzione collaborativa, distribuzione e manutenzione dei sistemi software, e conoscenza e comprensione degli strumenti che supportano queste attività.
- Conoscenza di come interpretare adeguatamente i principali passaggi degli algoritmi per l'IA.

Applicazione della conoscenza e della comprensione.

Al termine del corso lo studente sarà in grado di:

- Redigere la documentazione di un sistema software, analizzare le specifiche dei requisiti e pianificarne lo sviluppo.
- Impiegare un linguaggio di programmazione per sviluppare componenti software modulari e riutilizzabili.
- Partecipare alla produzione collaborativa, distribuzione e manutenzione dei sistemi software.
- Verificare la qualità dei componenti software e preparare la documentazione necessaria per il loro riutilizzo.
- Gestire il ciclo di sviluppo dei componenti software.
- Acquisire la capacità di utilizzare modelli computazionali per la soluzione di problemi classici di classificazione, clustering e regressione;
- Essere in grado di affrontare un problema di analisi dei dati (semplice) per sintetizzare nuova conoscenza implementando semplici sistemi decisionali (ad es. per prendere decisioni seguendo l'elaborazione di un segnale o di dati tabellari);
- Essere in grado di identificare le variabili che descrivono un problema decisionale con il massimo potere informativo
- Essere in grado di utilizzare gli strumenti software disponibili per l'applicazione dei metodi IA.

Giudizio

Le competenze di conoscenza e comprensione acquisite devono abilitare lo studente a valutare e selezionare, sulla base delle specifiche dei requisiti, gli strumenti e i componenti software più appropriati per lo sviluppo modulare e il riutilizzo del software in un ambiente distribuito. Inoltre, lo studente dovrebbe:

- Sapere giudicare quali sono le scelte appropriate da fare per risolvere casi applicativi reali.
- sapere giudicare le principali caratteristiche dei modelli computazionali presentati.
- sapere valutare l'adeguatezza di una procedura sperimentale.

Competenze comunicative

Lo studente svilupperà la capacità di comunicare, in modo preciso e competente, le scelte fatte nello sviluppo di applicazioni software, con particolare riferimento alla preparazione della documentazione volta al riutilizzo e alla manutenzione del software.

Inoltre, gli studenti dovrebbero essere in grado di redigere, presentare e spiegare possibili soluzioni progettuali a casi applicativi reali. Inoltre, dovrebbero essere in grado di spiegare i contenuti del corso in un linguaggio tecnico adeguato.

Competenze di apprendimento

Lo studente deve essere in grado di acquisire nuovi linguaggi e strumenti per lo sviluppo di componenti software modulari e riutilizzabili, nonché di identificare e utilizzare componenti software già disponibili.

Inoltre, gli studenti dovrebbero sviluppare le competenze di apprendimento necessarie per intraprendere successivi studi in temi di Intelligenza Artificiale con un alto grado di autonomia.

**L'attività didattica è offerta in:**

**Facoltà Dipartimentale di Ingegneria**

<b>Tipo corso</b>	<b>Corso di studio (Ordinamento)</b>	<b>Percorso</b>	<b>Crediti</b>	<b>S.S.D.</b>
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	15	ING-INF/05, ING-INF/05

*Stampa del 06/11/2025*

# Fondamenti di Robotica [ 2205106 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** LOREDANA ZOLLO

**Periodo:** Secondo Ciclo Semestrale

## Obiettivi formativi

Il corso si propone di fornire agli studenti una solida base teorica e pratica nel campo della robotica, preparandoli ad affrontare le sfide tecnologiche e scientifiche del settore. Il corso è concepito per fornire un'ampia panoramica dei concetti fondamentali della robotica, quali cinematica, dinamica e controllo di robot.

Il corso si propone inoltre di fornire agli studenti competenze pratiche sugli strumenti basati su intelligenza artificiale e apprendimento automatico per migliorare l'autonomia e l'adattabilità dei robot in ambienti dinamici e non strutturati.

Il corso si propone di fornire agli Studenti:

- (1) solide conoscenze teoriche per l'analisi, la modellazione, la programmazione e la configurazione di robot;
- (2) abilità riassunte come segue:
  - (i) basi pratiche sull'analisi, sviluppo e controllo di sistemi robotici;
  - (ii) competenze pratiche sugli strumenti per la modellazione, programmazione e gestione di robot.

## Prerequisiti

Nessuna.

Prerequisiti:

Fondamenti di informatica

## Contenuti del corso

- Introduzione alla robotica: concetti fondamentali, definizioni, principali strutture di manipolatori e componenti base ed esempi di applicazioni.
- Cinematica dei robot – Richiami di cinematica del corpo rigido; Posizione ed orientamento di un corpo rigido; Matrici di Rotazione; Angoli di Eulero; Asse/angolo e quaternione; Convenzione di Denavit-Hartenberg; Cinematica diretta.
- Cinematica differenziale e statica – Cinematica differenziale; Jacobiano geometrico e jacobiano analitico; Singolarità cinematiche; Ridondanza; Cinematica inversa e algoritmi di inversione cinematica; Statica.
- Pianificazione di traiettoria – Traiettorie nello spazio dei giunti; traiettorie punto-punto nello spazio operativo; traiettorie generate con algoritmi di intelligenza artificiale.
- Tecniche ad apprendimento per pianificazione e controllo dei robot, quali apprendimento per dimostrazione, ottimizzazione e apprendimento per rinforzo.
- Applicazioni dell'intelligenza artificiale alla modellazione cinematica e dinamica dei robot; tecniche di machine learning per la visione artificiale
- Introduzione alla programmazione in Robot Operating System (ROS2) – Sistemi di messaggistica e gestione di pacchetti (es. topic, service ed action); modellazione dinamica dei robot tramite URDF; controlli in ROS2; Visualizzazione di robot in Rviz; Simulazione di robot in Gazebo.

## Metodi didattici

- Lezioni frontali (48 ore), in cui vengono presentati gli argomenti del corso e svolti esercizi che ne mostrano l'applicazione a problemi specifici.
- Seminari (4 ore) su specifiche applicazioni di robotica autonoma.
- Esercitazioni in aula e laboratori didattici (20 ore) per insegnare l'uso degli strumenti software necessari per lo sviluppo ed il controllo dei robot (ambiente ROS2 in linguaggio Python).
- Progetti di gruppo nei quali gli studenti metteranno in pratica gli insegnamenti appresi durante le lezioni frontali. I gruppi saranno composti da un massimo di 4 studenti e dovranno occuparsi della modellazione e dello sviluppo in ambiente di simulazione di un sistema robotico autonomo per l'esecuzione di un compito specifico definito all'inizio del corso.

## Modalità di verifica dell'apprendimento

Le conoscenze relative al corso sono verificate mediante prova orale e presentazione di progetti che prevedono l'uso del framework ROS2 (Robot Operating System).

La valutazione delle conoscenze acquisite verrà effettuata dai docenti, che verificheranno l'apprendimento delle conoscenze teoriche oggetto del corso, e dai tutor del corso, ai quali gli studenti dovranno mostrare, tramite la discussione del progetto realizzato, la loro capacità di applicare le conoscenze teoriche ad un problema pratico.

Durante la prova orale i docenti faranno tre domande, in forma scritta o verbale, volte ad accertare la conoscenza teorica da parte dello studente degli argomenti trattati a lezione. Le tre domande sono valutate con uguale peso (11 punti). La prova orale è sia a stimolo chiuso che a stimolo aperto e con risposta aperta.

La presentazione del progetto consiste in una illustrazione dei metodi applicati e dei risultati raggiunti tramite l'utilizzo di slide. Tutti i progetti verranno presentati e discussi alla fine del corso o comunque prima del primo appello d'esame. In sede di discussione dei progetti verranno poste domande specifiche a ciascuno studente per valutare l'apporto individuale al lavoro complessivo.

La prova orale e la discussione dei progetti mirano ad accertare:

1. Conoscenze e capacità di comprensione degli argomenti del corso;
2. Capacità di applicare le conoscenze e competenze acquisite nella formulazione di soluzioni (anche originali) a problemi di progettazione e sviluppo di robot;
3. Capacità di applicare i metodi e gli strumenti presentati durante il corso per la risoluzione di problemi di modellazione e controllo dei robot;
4. Abilità comunicative nella descrizione formale degli argomenti del corso, esprimendo chiaramente e senza ambiguità le proprie conclusioni, e le conoscenze e le considerazioni che le sottendono, a interlocutori specialisti e non specialisti in contesti nazionali e internazionali;
5. Autonomia di giudizio nella scelta delle soluzioni a problemi di progettazione e sviluppo di robot e capacità di gestire attività o progetti tecnici o professionali complessi nel loro campo di studio e di assumersi la responsabilità di prendere decisioni;
6. Capacità di impegnarsi in modo autonomo nell'apprendimento permanente.

E' prevista una durata massima della prova orale di 45 minuti e una durata massima della presentazione dei progetti di 30 minuti.

La valutazione finale viene formulata secondo la seguente regola:

- 8/9 del voto finale assegnato mediante la prova orale;
- 1/9 del voto finale assegnato mediante la presentazione dei progetti.

L'esame è superato se il candidato raggiunge almeno i 18/30.

La lode viene attribuita agli studenti che abbiano conseguito il punteggio massimo su tutte le prove raggiungendo un punteggio finale superiore a 30/30.

### **Testi di riferimento**

- B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics - Modelling, Planning and Control, Springer 2010.
- Rico, Francisco Martín. A concise introduction to robot programming with ROS2. Chapman and Hall/CRC, 2022.
- Aude Billard, Sina Mirrazavi and Nadia Figueroa - Learning for Adaptive and Reactive Robot Control". MIT Press, 2022
- H. Govers. Artificial Intelligence for Robotics - Second Edition: Build intelligent robots using ROS 2, Python, OpenCV, and AI/ML techniques for real-world tasks, Packt.
- Dispense e materiali didattici forniti dal docente.

### **Altre informazioni**

Conoscenza e capacità di comprensione

- Principi alla base dei sistemi robotici, degli schemi di funzionamento e delle metodiche di modellazione, programmazione e configurazione;
- Strumenti software di ausilio alla modellazione, programmazione e gestione di sistemi robotici in ambienti dinamici.

Conoscenze e capacità di comprensione applicate:

- Capacità di applicare metodi e strumenti di elettronica, informatica, automatica e meccanica all'analisi di sistemi complessi quali i robot;
- Capacità di sviluppare, con metodiche avanzate, sistemi che integrano il mondo fisico (inclusa la componente umana) con tecniche computazionali, metodi di controllo, tecnologie informatiche e di comunicazione, proponendo anche soluzioni innovative per singoli componenti o per il sistema integrato;
- Capacità di utilizzare gli strumenti software di ausilio alla modellazione, programmazione e gestione di sistemi robotici per uso nei processi industriali e di robotica collaborativa in contesti di interazione e cooperazione con l'uomo.

Autonomia di giudizio: Gli studenti saranno stimolati allo sviluppo delle proprie capacità analitiche e critiche tramite la proposizione di esercizi e di attività pratiche su tematiche trattate in aula.

Capacità di apprendimento: il corso persegue un approccio di coinvolgimento attivo dello studente nel proprio percorso formativo, stimolando la rivisitazione e l'approfondimento di competenze acquisite negli studi precedenti, e l'applicazione dei concetti appresi ad ambiti specifici.

Abilità comunicative e soft skill: L'insegnamento si propone inoltre di sviluppare abilità relative alla sfera delle abilità comunicative e delle soft-skill per operare in team e in contesti multidisciplinari. Tale obiettivo sarà perseguito

cercando di promuovere il coinvolgimento proattivo degli studenti durante le ore di didattica frontale e attraverso la conduzione di attività di gruppo tese allo svolgimento di attività progettuali semplici che richiedono la messa in pratica delle nozioni teoriche apprese.

**L'attività didattica è offerta in:**

**Facoltà Dipartimentale di Ingegneria**

<b>Tipo corso</b>	<b>Corso di studio (Ordinamento)</b>	<b>Percorso</b>	<b>Crediti</b>	<b>S.S.D.</b>
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	9	ING-IND/34

*Stampa del 06/11/2025*

# Inglese Generale [ 22051C1 ]

**Offerta didattica a.a. 2025/2026**

**Docenti:** ROBERTA ARONICA, DOCENTE\_FITTIZIO DOCENTE\_FITTIZIO

**Periodo:** Primo Ciclo Semestrale

## Obiettivi formativi

Il corso è finalizzato al consolidamento del livello B2 CEFR in preparazione al livello C1 CEFR. Le attività didattiche sono impartite da docenti madrelingua che collaborano con il Centro linguistico di Ateneo.

## Prerequisiti

Gli studenti in possesso di certificazioni linguistiche – rilasciate da non più di tre anni e da uno dei seguenti Enti Certificatori: Cambridge Assessment English; LinguaSkill; City and Guilds, Pitman; Edexcel / Pearson Ltd; IELTS; TCL Trinity College London; TOEFL ET – di livello pari o superiore al C1 CEFR possono ottenere l'esonero previa domanda all'attenzione del Centro Linguistico d'Ateneo (cla@unicampus.it).

## Contenuti del corso

Nel corso curricolare semestrale da 3 CFU si approfondiscono le strutture logico-grammaticali e il vocabolario della lingua inglese al fine di consolidare il livello B2 CEFR e iniziare il percorso di preparazione al C1 CEFR.

## Metodi didattici

Il corso viene erogato in aula attraverso lezioni frontali ed esercitazioni.

## Modalità di verifica dell'apprendimento

Esame di idoneità.

La verifica dell'apprendimento viene effettuata attraverso una prova scritta composta da esercizi di grammatica, comprensione del testo, scrittura e ascolto.

Le conoscenze lessicali e grammaticali e le abilità relative alla comprensione e alla produzione scritta sono verificate mediante una prova scritta e una di ascolto con rispettivo test di comprensione a risposta aperta di livello commisurato all'obiettivo individuale. Le abilità comunicative (speaking) vengono valutate dal docente durante il corso attraverso attività interattive. Il risultato della prova è espresso come giudizio di idoneità. Per conseguire l'idoneità lo studente dovrà ottenere un punteggio totale uguale o maggiore al 60%.

## Testi di riferimento

Il materiale didattico viene fornito dai docenti.

## Altre informazioni

Ogni studente è tenuto a sostenere un test di posizionamento per individuare il livello iniziale di conoscenza della lingua inglese.

Capacità di applicare conoscenza e comprensione

Al termine del corso, lo studente dovrà essere in grado di:

- dialogare, leggere e scrivere brani in lingua inglese, rispondere a domande di comprensione del testo;
- produrre un testo scritto di argomento generale.

Autonomia di giudizio

Lo studente sarà stimolato a sviluppare un approccio critico sulla propria capacità di comprensione del testo ascoltato e sulla elaborazione di un testo in inglese usando gli strumenti che l'insegnante proporrà con gradualità durante il corso. Gli studenti saranno sollecitati alla verifica autonoma sia attraverso la correzione di propri elaborati che di verifica sul livello di comprensione dei testi analizzati durante le lezioni frontali.

Abilità nella comunicazione

Lo studente dovrà essere in grado di comunicare in modo chiaro e grammaticalmente corretto.

Capacità di apprendere

Lo studente dovrà dimostrare una partecipazione attiva interagendo in lingua inglese con l'insegnante e con l'aula.

**L'attività didattica è offerta in:**

**Facoltà Dipartimentale di Ingegneria**

<b>Tipo corso</b>	<b>Corso di studio (Ordinamento)</b>	<b>Percorso</b>	<b>Crediti</b>	<b>S.S.D.</b>
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	3	L-LIN/12

*Stampa del 06/11/2025*



# **Modelli e Metodi di Ottimizzazione e Statistica [ 2205101 ]**

**Offerta didattica a.a. 2025/2026**

**Docenti:** MARCO PAPI

**Periodo:** Primo Ciclo Semestrale

## **Obiettivi formativi**

L'obiettivo del corso è quello di fornire allo studente conoscenze e competenze di base sia operative sia metodologiche riguardanti l'ottimizzazione e la statistica con particolare enfasi ai modelli ed algoritmi per le applicazioni ingegneristiche ed informatiche in ambito industriale e gestionale. Lo studente è inoltre introdotto all'analisi dei dati, al ragionamento probabilistico e all'inferenza statistica, mostrando come l'uso di opportuni metodi statistici permetta di risolvere una varietà di problemi concreti a partire dall'analisi dei dati. Lo studio teorico dei principali algoritmi per il calcolo della soluzione ottima è completato dalla sperimentazione numerica mediante linguaggi di modellazione.

I contenuti riguardanti l'ottimizzazione e la statistica sono finalizzati a fornire i concetti, sia di carattere modellistico sia algoritmico, utilizzati in numerosi contesti applicativi, relativamente ai problemi decisionali strutturati presenti nella fase di progettazione e/o gestione di un sistema.

## **Prerequisiti**

Conoscenze di base di analisi matematica e algebra lineare, elementi di calcolo delle probabilità e statistica.

## **Contenuti del corso**

1) Ricerca operativa (42 ore circa): Formulazione di un modello di programmazione matematica: variabili decisionali, funzione obiettivo, vincoli. Tecniche di modellazione matematica. Problemi su grafi e reti di flusso. Alberi e grafi, algoritmi di ricerca su grafo. Cammini minimi, massimo flusso, flusso di costo minimo, assegnamento. Alcune strutture dati e algoritmi di soluzione. Programmazione lineare. Metodo grafico. Soluzioni di base e condizioni di ottimalità. Metodo del simplesso. Teoria della dualità, coppie di problemi duali, scarti complementari. Metodo del simplesso duale. Analisi di sensitività, analisi parametrica. Programmazione lineare mista intera e ottimizzazione combinatoria. Problemi di ottimizzazione discreta: formulazioni. Rilassamenti e algoritmo di Branch-and-Bound. Alcune applicazioni. Problemi di ottimizzazione convessa e concava. Ottimizzazione vincolata (su insiemi convessi e su un poliedro). Algoritmi per la risoluzione di problemi con vincoli convessi. Le condizioni di Karush-Kuhn-Tucker. Metodi euristici e di ricerca locale. Applicazioni reali in ambito industriale e in computer science.

2) Statistica e analisi decisionale (30 ore circa): Richiami di statistica descrittiva. Frequenze assolute, relative e cumulate. Indici statistici di posizione e di variabilità. Misure di dispersione. Indipendenza e dipendenza tra variabili quantitative. Variabili casuali discrete e continue. Distribuzioni e principali modelli probabilistici. Metodi di stima dei parametri. Regressione semplice e multipla: modello statistico ed assunzioni. Approcci analitici alla regressione. Analisi fattoriale. Analisi della varianza. Analisi per componenti principali (PCA) e linear discriminant analysis (LDA). Certezza, rischio ed incertezza. Condizioni di rischio. Decisioni strutturate e non strutturate. Decisioni in condizioni di incertezza. Alternative, outcomes e stati della natura, matrice di decisione. Criterio del valore atteso. Elementi di teoria bayesiana della decisione. La funzione di verosimiglianza. Il metodo bayesiano.

## **Metodi didattici**

- Lezioni frontali (52 ore) in cui verranno presentati gli argomenti del corso e svolti esercizi guida che ne mostrano l'applicazione a problemi specifici.
- Esercitazioni (20 ore) con cadenza settimanale durante il periodo di erogazione del corso e in preparazione della prova d'esame.

## **Modalità di verifica dell'apprendimento**

L'esame ha lo scopo di verificare se il candidato ha acquisito le conoscenze e le abilità specificate nei risultati di apprendimento attesi. L'esame consiste nello svolgimento di una prova scritta e di un elaborato di tipo progettuale. La prova scritta riguarda esercizi di tipo pratico inerenti ai contenuti del corso. Nello specifico la prova scritta è composta da tre esercizi di cui i primi due riguardano il punto 1) del programma mentre il terzo esercizio riguarda il punto 2) del programma. Il punteggio massimo conseguibile nella prova scritta è pari a 32. La prova ha una durata di 2.5 ore.

L'elaborato progettuale riguarda un argomento scelto dallo studente nell'ambito di una lista fornita dal docente oppure una tematica di elezione da parte dello studente e concordata con il docente. In tale progetto, lo studente deve dimostrare di sapere applicare le metodologie studiate durante il corso per la soluzione del problema scelto, mettendone in luce pro e contro e giustificando le proprie scelte implementative. Il punteggio massimo attribuito all'elaborato è pari a 30. La valutazione dell'apprendimento prevede l'attribuzione di un voto finale dato dall'esito combinato del punteggio della prova scritta (60% del voto finale) e del punteggio assegnato al progetto svolto (40% del voto finale). L'esame si considera superato qualora lo studente consegua un punteggio maggiore o uguale a 18/30. Il voto finale è espresso in 30-esimi. Lo studente che consegue un punteggio complessivo superiore a 30 consegue la lode.

### Testi di riferimento

- [1] F. S. Hillier, G. J. Lieberman, Ricerca Operativa: Fondamenti, Editore: McGraw-Hill.
- [2] Fletcher, R. Practical: Methods of Optimization, Wiley.
- [3] W.L. Winston, Operations Research, Applications and Algorithms, third edition.
- [4] Dimitri P. Bertsekas, Nonlinear Programming, Athena Scientific.

### Altre informazioni

Al termine del corso lo studente è in grado di:

- a. Individuare i metodi da utilizzare per la determinazione della soluzione ottima di un problema di ricerca operativa, conoscendone applicabilità e limiti.
- b. Analizzare la soluzione ottima di un problema determinata dal metodo risolutivo, in particolare la sua unicità e la sua sensibilità rispetto ai parametri del modello, sulla base dei dati disponibili.
- c. Utilizzare l'ambiente di modellazione per la codifica dei modelli formulati e la loro risoluzione.
- d. Formulare un problema decisionale in un contesto reale mediante un modello di programmazione matematico-statistico, individuando le variabili decisionali ed esprimendo, in funzione di esse, l'obiettivo da conseguire ed i vincoli che devono essere rispettati affinché la soluzione sia effettivamente utilizzabile nel contesto reale.
- e. Applicare i fondamenti metodologici dell'analisi dei dati.
- f. Utilizzare le principali tecniche statistiche per l'analisi di dati.

**L'attività didattica è offerta in:**

### Facoltà Dipartimentale di Ingegneria

Tipo corso	Corso di studio (Ordinamento)	Percorso	Crediti	S.S.D.
Corso di Laurea Magistrale	Ingegneria dei Sistemi Intelligenti (2025)	comune	9	SECS-S/06

Stampa del 06/11/2025