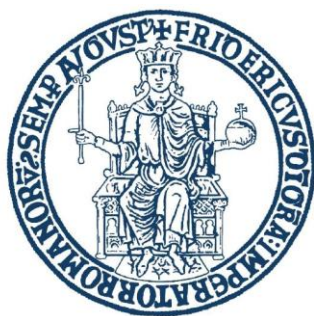




UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II

SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

DIPARTIMENTO DI INGEGNERIA ELETTRICA
E DELLE TECNOLOGIE DELL'INFORMAZIONE

GUIDA DELLO STUDENTE

CORSO DI LAUREA MAGISTRALE IN INFORMATICA

Classe delle Lauree Magistrali in Informatica- LM-18

ANNO ACCADEMICO 2025/2026

Vers. 1.2

AGGIORNATA: 29/08/2025

Sommario

Revisioni.....	3
Introduzione	4
Requisiti d'ingresso e attività formative propedeutiche e integrative.....	6
Obiettivi e finalità del Corso di Laurea	7
Sbocchi occupazionali e professionali previsti per i laureati.....	8
Regolamento vigente.....	9
Piani di Studio	10
Manifesto degli Studi del Corso di Laurea Magistrale in Informatica	13
Descrizione dei percorsi.....	18
Programma Erasmus.....	25
Esame finale e attribuzione del punteggio finale	26
Calendario Attività Didattiche	27
Referenti del Corso di Studi	28
Contatti e Strutture	29
Schede Degli Insegnamenti.....	30

Per ulteriori informazioni aggiornate si veda <http://informatica.dieti.unina.it/>.

Revisioni

Data Aggiornamento	Versione	Descrizione
22/07/2025	1.0	Prima versione della guida 2025/2026.
29/08/2025	1.1	Aggiornato i requisiti e le modalità per l'accesso
29/08/2025	1.2	Aggiunta scheda dell'insegnamento per il nuovo insegnamento a scelta libera di "Generative AI"

Introduzione

Il Corso di Laurea Magistrale in Informatica dell'Università di Napoli Federico II da oltre 20 anni mira a fornire competenze avanzate e costantemente aggiornate nei principali ambiti teorici, metodologici e tecnologici dell'Informatica, atte a sviluppare le capacità di problem solving ed innovazione dello studente.

Grazie ad opportune competenze teoriche e metodologiche, affiancate e completate da aspetti tecnologici avanzati costantemente aggiornati, i laureati saranno in grado di:

- aggiornarsi indipendentemente nel corso della propria carriera (formazione permanente) in una disciplina, quale l'informatica, soggetta a continua evoluzione;
- trovare ampi sbocchi professionali con responsabilità tecnico/decisionali, sia in Italia che all'estero, in contesti quali Software House e Aziende di ICT, Libera Professione, Start-up, Ricerca e Sviluppo nelle grandi aziende, Enti e Amministrazioni pubbliche, Centri di Ricerca e Università, Insegnamento.

L'attività formativa è finalizzata a fornire agli studenti capacità di problem solving e progettazione del software nei principali contesti applicativi dell'informatica. La finalità formativa è conseguita prevedendo sia insegnamenti riguardanti fondamenti logico-matematici, sia approfondimenti metodologici e tecnologici nei temi tecnologici chiave dell'informatica moderna.

I diversi insegnamenti erogati consentono di articolare percorsi di studio strutturati sulle seguenti aree di forte interesse:

- Artificial Intelligence
- Data Science
- Security and Privacy
- Computer games
- Complex Software Systems

Lo studente può personalizzare il proprio percorso di studi integrando gli insegnamenti obbligatori con opportune scelte tra gli insegnamenti facoltativi a scelta libera e vincolata. Il dettaglio per la strutturazione del piano di studi è descritto nelle sezioni successive del presente documento.

Il laureato magistrale in Informatica, previo esame di stato, può iscriversi all'Albo degli Ingegneri dell'Informazione, sez. A ottenendo quindi il titolo di *Ingegnere dell'Informazione*.

Il Corso di Laurea Magistrale in Informatica ottiene regolarmente il *Bollino GRIN*. Il [Bollino GRIN](http://www.grin-informatica.it/opencms/opencms/grin/didattica/bollino.html), Tale riconoscimento, erogato in collaborazione tra GRIN (Gruppo di Informatica - l'associazione italiana dei professori universitari di informatica) e AICA (Associazione Italiana per l'Informatica ed il Calcolo Automatico), certifica la qualità dei contenuti delle lauree triennali e magistrali di informatica (classi L-31 e LM-18). Maggiori informazioni sulla certificazione GRIN possono essere reperite all'indirizzo:

<http://www.grin-informatica.it/opencms/opencms/grin/didattica/bollino.html>

Anche dalle indagini *AlmaLaurea* emerge la forte competitività di questo CdS rispetto ai suoi omologhi di altre sedi, in termini di livelli occupazionali, attesa del primo impiego, percentuale di contratti a tempo indeterminato, livelli di retribuzione, soddisfazione degli studenti e utilità percepita.

I Corsi di Laurea e di Laurea Magistrale in Informatica sono gestiti dal Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione (DIETI). Il DIETI ha ottenuto dal Ministero della Ricerca il riconoscimento di *Dipartimento di Eccellenza* per il quinquennio 2023/2027.

Per ulteriori dettagli far riferimento al sito web del corso di studi: <https://informatica.dieti.unina.it/>

Requisiti d'ingresso e modalità di accesso

Il Corso di Laurea Magistrale in Informatica è ad accesso libero. L'immatricolazione si effettua attraverso **SegrePass** (<https://www.segrepass2.unina.it/Welcome.do>), il sistema di Ateneo che permette di accedere ai servizi per gli studenti.

Conoscenze richieste, requisiti curricolari e dettagli sulle modalità di accesso al Corso di Studi Magistrale sono dettagliati sul sito web del corso di studi al seguente indirizzo:

- <https://informatica.dieti.unina.it/index.php/it/laurea-magistrale/requisiti-e-modalita-di-accesso>

Obiettivi e finalità del Corso di Laurea

Il Corso di Laurea Magistrale in Informatica si pone come obiettivo l'integrazione ed il rafforzamento del processo formativo di base intrapreso nel I ciclo attraverso un ordinamento che si adatti con la massima flessibilità alle esigenze formative dello studente e alle richieste delle varie parti interessate, in particolare il mercato del lavoro. Pertanto, in accordo con le linee guida delle associazioni nazionali (GRIN) ed internazionali (ACM/IEEE) del settore, il percorso didattico prevede la formazione di solide conoscenze sia dei fondamenti che degli aspetti applicativi dei vari settori dell'informatica.

Obiettivi comuni a tutti i percorsi di studio interni al corso di laurea sono:

- l'acquisizione del metodo scientifico di indagine che prevede l'utilizzazione degli strumenti matematici che sono di supporto all'informatica ed alle sue applicazioni;
- l'approfondimento delle tecnologie dei sistemi di elaborazione e gestione dell'informazione;
- l'approfondimento delle metodologie di progettazione e realizzazione dei sistemi informatici;
- l'approfondimento di specifici settori di applicazione dei sistemi informatici e delle tecnologie informatiche;
- l'acquisizione di elementi di cultura aziendale e professionale.

Sbocchi occupazionali e professionali previsti per i laureati

Gli ambiti occupazionali e professionali di riferimento per i laureati magistrali della classe sono quelli della progettazione, organizzazione e manutenzione di sistemi informatici orientati anche alla gestione di sistemi complessi o innovativi (con specifico riguardo ai requisiti di affidabilità, prestazioni e sicurezza), sia in imprese produttrici nelle aree dei sistemi informatici e delle reti, sia nelle imprese e nelle pubbliche amministrazioni.

L'alta qualità del corso di studi magistrale in Informatica della Federico II è riconosciuta dal mondo del lavoro: **già a un anno dalla Laurea il tasso di disoccupazione dei suoi laureati è nullo (fonte Almalaurea, indagine 2022).**

La richiesta di laureati supera costantemente l'offerta, il che permette di scegliere il contesto e la nazione in cui lavorare.

Inoltre, i laureati magistrali in Informatica possono accedere all'Esame di Stato per l'iscrizione all'Albo degli Ingegneri.

I laureati magistrali potranno trovare impiego in aziende produttrici di software innovativo e in centri di ricerca e sviluppo pubblici e privati, nonché in aziende, enti e organismi che offrono servizi informatici avanzati.

Nel quadro di riferimento fornito dalla Classificazione delle Professioni dell'ISTAT, Edizione 2001, parte seconda, tali occupazioni ricadono nel settore 2- "Professioni intellettuali, scientifiche e di elevata specializzazione", Specialisti in scienze matematiche, fisiche, naturali ed assimilati, gruppo 2.1.1.4 "Informatici e telematici".

Il corso prepara alle professioni di:

- Analisti e progettisti di software applicativi e di sistema
- Analisti di sistema
- Specialisti in sicurezza informatica
- Specialisti in reti e comunicazioni informatiche
- Specialisti nella ricerca informatica di base

Regolamento vigente

Nell'anno accademico 2020/21 la laurea magistrale ha subito una variazione di regolamento che ne ha reso più moderni la struttura e i contenuti. Il corrente anno accademico è conforme a tale regolamento aggiornato. In particolare, ciascuno studente segue un corso di studi conforme al regolamento vigente nell'anno della sua immatricolazione e, dunque gli studenti immatricolati dall'a.a. 2020/21 in poi seguono il regolamento attualmente vigente. Gli studenti immatricolati precedentemente seguono un piano di studi articolato secondo il regolamento in vigore al momento della loro iscrizione.

Salvo indicazione contraria, le informazioni di seguito riportate fanno riferimento al regolamento della laurea in vigore dall'a.a. 2020-21.

I regolamenti del Corso di Studi possono essere recuperati nel sito del corso di studi all'indirizzo:

<https://informatica.dieti.unina.it/index.php/it/laurea-magistrale/regolamento-laurea-magistrale>

Piani di Studio

Lo studente compone il suo piano di studi integrando il nucleo degli insegnamenti **obbligatori** con due tipologie di scelte: insegnamenti a **scelta vincolata** e insegnamenti a scelta libera.

Gli insegnamenti a scelta vincolata determinano la scelta del **percorso tematico**. Il percorso tematico permette di indirizzare lo studio verso un'area specializzata di interesse. I percorsi previsti sono i seguenti

- Artificial Intelligence
- Data Science
- Security and Privacy
- Computer games
- Complex Software Systems

La struttura degli insegnamenti per anno è la seguente:

Insegnamento o attività formativa	Modulo	CFU
I anno		
Logic for computer science		6
Advanced databases	DB technologies	6
	NoSQL	6
Machine Learning	Neural networks and deep learning	6
	Statistical learning	6
Computational complexity		6
Operating systems for mobile, cloud and IoT		6
Parallel and distributed computing (*)		6
<i>Esami a libera scelta (Tab. B)</i>		6
<i>Esami a scelta vincolata (Tab. A)</i>		6
II anno		
Intelligent web		6
Operations research (**)		6
<i>Esami a scelta vincolata (Tab. A)</i>		6
<i>Esami a libera scelta (Tab. B)</i>		6
Altre attività formative		1
Prova finale		29

Note:

(*) Se già sostenuto alla triennale (o sostenuto con la titolatura Calcolo parallelo e distribuito Mod.A), sostituzione obbligatoria con *“Parallel high performance computing: tools, algorithms and software”*

(**) Se già sostenuto alla triennale (o sostenuto con la titolatura Ricerca operativa) sostituzione obbligatoria con *Combinatorial optimization*.

Le eventuali sostituzioni vanno comunicate al Coordinatore del corso di studi all’inizio dell’anno accademico (preferibilmente entro il 31 dicembre 2025 o all’atto dell’iscrizione se avviene successivamente).

La scelta del percorso tematico impone il conseguimento di **entrambi** gli insegnamenti associati ad un percorso. Lo studente potrà scegliere tra uno dei seguenti percorsi.

Percorso	Insegnamenti	A n n o	C F U
Artificial Intelligence	Methods for artificial intelligence	I	6
	Computer vision	II	6
Data science	Data analytics	I	6
	Algorithm design	II	6
Security and privacy	Biometric systems	I	6
	Security and privacy	II	6
Computer games	Game design and development	I	6
	Computer graphics	II	6
Complex software systems	Software project management and evolution	I	6
	Automated software verification	II	6

I 18 CFU legati agli esami a libera scelta possono essere conseguiti scegliendo liberamente insegnamenti attivati presso l’Ateneo:

- Se tali CFU sono conseguiti attingendo esclusivamente alla lista di esami attivati per il Corso di Laurea Magistrale in Informatica (Tabella B del Manifesto degli Studi, a pagina 13), lo studente **NON** è tenuto a presentare un piano di studi preventivo.
- Se tali CFU sono conseguiti avvalersi anche in parte con insegnamenti erogati dall’Ateneo ma non attivati per il Corso di Laurea Magistrale in Informatica, è necessario presentare un piano di studi al Coordinatore del Corso di Studi rispettando i tempi e le modalità descritti nella sezione dedicata alla comunicazione del piano di studi ed in ogni caso PRIMA del conseguimento dell’esame.

Non è necessario presentare un piano di studi per comunicare la scelta tematica (gli insegnamenti a scelta vincolata). Lo studente sceglierà autonomamente e coerentemente gli esami. La registrazione degli esami determinerà implicitamente la scelta del percorso.

In assenza di presentazione del piano di studi, la Segreteria Didattica al momento della registrazione degli esami convaliderà nella carriera i soli esami attivati per il corso di studi elencati nella Tabella B. Pertanto, è nella responsabilità dello studente una scelta coerente degli insegnamenti sia a scelta libera che vincolata. Scelte non coerenti non permetteranno la registrazione degli esami connessi eventualmente sostenuti.

Manifesto degli Studi del Corso di Laurea Magistrale in Informatica

In ogni anno accademico viene approvato il *Manifesto degli Studi*, che formalmente struttura l'offerta formativa per gli studenti che si immatricolano in quell'anno accademico. Il Manifesto degli Studi definisce obbligatorietà, propedeuticità, e attivazioni degli insegnamenti facoltativi. In assenza di cambiamento di regolamento gli insegnamenti obbligatori e a scelta vincolata rimangono immutati in ogni manifesto. **Quello che può variare di anno in anno è invece l'attivazione degli insegnamenti a libera scelta (tabella B). In tutta la sua carriera, lo studente deve fare riferimento al Manifesto degli Studi dell'anno della sua immatricolazione.**

Di seguito viene riportato il Manifesto degli Studi approvato dall'A.A. 2024-2025 (riferimento per gli immatricolati nell'A.A. 2025-2026).

Manifesto degli Studi del Corso di Laurea in Informatica

Classe delle lauree in Informatica – Classe LM-18

A.A. 2025-2026

Insegnamento o attività formativa	Modulo	Semestre	CFU	SSD	Tipologia (*)	Propedeuticità
I anno						
Advanced databases	DB Technologies	1	6	INFO-01/A	2	
	NoSQL	2	6	INFO-01/A		
Machine Learning	Neural networks and deep learning	1	6	INFO-01/A	2	
	Statistical Learning	2	6	INFO-01/A		
Operating systems for mobile, cloud and IoT		1	6	INFO-01/A	2	
Parallel and distributed computing (*)		1	6	INFO-01/A	2	
Computational complexity		2	6	INFO-01/A	2	
Logic for computer science		2	6	INFO-01/A	4	
Esami a libera scelta (vedi Tab.B)			6		3	
Esami a scelta vincolata (Tab. A)			6		2	
II anno						
Intelligent web		1	6	INFO-01/A	2	
Operation research (**)		1	6	MATH-06/A	4	
Esami a scelta vincolata (Tab. A)			6		2	
Esami a libera scelta (vedi Tab.B)			12		3	
Altre attività formative			1		6	
Prova finale			29		5	

Note:

(*) Se già sostenuto alla triennale, sostituzione obbligatoria con “[Parallel High Performance Computing: tools, algorithms and software](#)”

(**) Se già sostenuto alla triennale, sostituzione obbligatoria con [Combinatorial optimization](#).

(*) Legenda delle tipologie delle attività formative ai sensi del DM 270/04

Attività formativa	1	2	3	4	5	6	7
rif. DM270/04	Art. 10 comma 1, a)	Art. 10 comma 1, b)	Art. 10 comma 5, a)	Art. 10 comma 5, b)	Art. 10 comma 5, c)	Art. 10 comma 5, d)	Art. 10 comma 5, e)

Tabella A: Esami a scelta vincolata

Scelta	Insegnamenti	Modulo	SSD	CFU	Semestre	Anno
Artificial Intelligence	Methods for Artificial Intelligence		INFO-01/A	6	2	I
	Computer vision		INFO-01/A	6	1	II
Data science	Data analytics		IINF-03/A	6	1	I
	Algorithm design		INFO-01/A	6	2	II
Security and privacy	Biometric systems		INFO-01/A	6	2	I
	Security and privacy		INFO-01/A	6	1	II
Computer games	Game design and development		INFO-01/A	6	1	I
	Computer graphics		INFO-01/A	6	1	II
Complex software systems	Software project management and evolution		INFO-01/A	6	2	I
	Automated Software Verification		INFO-01/A	6	2	II

Tabella B: Esami a libera scelta coerenti con gli obiettivi formativi del Corso di Studi (i cui CFU sono pienamente riconosciuti senza previa delibera della Commissione di Coordinamento Didattico)

Materie a scelta Tabella B	SSD	CFU	Semestre	Propedeuticità	Mutuazione
Algorithm design	INFO-01/A	6	2		
Automated software verification	INFO-01/A	6	2		
Bioinformatics	INFO-01/A	6	2		
Biometric systems	INFO-01/A	6	2		
Calcolo numerico	MATH-05/A	6	2		LT informatica L31
Combinatorial optimization	MATH-06/A	6	2	Operations research	
Computer forensics	INFO-01/A	6	2		
Computer graphics	INFO-01/A	6	1		
Computer vision	INFO-01/A	6	1		
Data analytics	IINF-03/A	6	1		
Didattica dell'Informatica	INFO-01/A	6	1		
Diritto dell'Informatica	GIUR-17/A	6	1		
Encoding and encryption	INFO-01/A	6	1		
Formal methods for strategic reasoning	INFO-01/A	6	2		
Game design and development	INFO-01/A	6	1		
Game engines and interactive experience	INFO-01/A	6	2		
Human-Robot interaction	INFO-01/A	6	1		
Intelligent robotics	INFO-01/A	6	2		
Istituzioni di matematica II	MATH-03/A	6	1		LT ottica e optometria
Linguaggi di programmazione II	INF/01	6	2		LT informatica L31
Mathematics for cryptography	MAT-03/A	6	1		
Methods for Artificial Intelligence	INFO-01/A	6	2		
Multimedia information systems	INFO-01/A	6	2		
Natural language processing	INFO-01/A	6	2		
Network security	IINF-05/A	6	1		LM-32 ING-INF
Parallel High Performance Computing: tools, algorithms and software	INFO-01/A	6	2		
Scientific computing	MATH-05/A	6	1		
Secure systems design	IINF-05/A	6	1		LM-32 ING-INF
Security and privacy	INFO-01/A	6	1		
Social, ethical, and psychological issues in AI	INFO-01/A	6	1		
Software project management and evolution	INFO-01/A	6	2		
Software testing	IINF-05/A	6	1		LM-32

					ING-INF
Tecnologie Web	INFO-01/A	6	2		LT informatica

Descrizione dei percorsi

Nella seguente tabella viene riportata la corrispondenza tra i **percorsi** dei piani di studio e gli insegnamenti obbligatori, a scelta vincolata e a scelta libera. La corrispondenza delineata permette di avere un quadro complessivo più completo dell'offerta formativa rispetto ad ogni percorso. Lo studente che desiderasse strutturare il suo piano di studio in modo perfettamente centrato sul tema prescelto può esercitare la scelta autonoma (per 18 CFU) negli insegnamenti associati al tema. Dopo la tabella di corrispondenza viene riportata una descrizione generale dei percorsi. Nelle corrispondenze ai percorsi sono riportati anche gli insegnamenti obbligatori per avere una panoramica complessiva dell'approfondimento sul tema. Per i contenuti dei corsi lo studente può far riferimento alla sezione dedicata alle schede dei corsi. Di seguito per ciascun tema viene riportata una breve descrizione e suggerimenti per gli esami a scelta.

Percorso	Obbligatori	Vincolati	Consigliati
Artificial Intelligence	Machine Learning (Statistical learning) Machine Learning (Neural networks and deep learning) Logic for computer science Intelligent Web	Methods for artificial intelligence Computer vision	Intelligent robotics Human-robot interaction Natural language processing Social, ethical, and psychological issues in artificial intelligence Formal methods for strategic reasoning
Data Science	Machine Learning (Statistical learning) Machine Learning (Neural networks and deep learning) Advanced databases (DB Technologies) Advanced databases (Mod. NoSQL)	Algorithm design Data analytics	Combinatorial optimization Biometric systems Bioinformatics Computer vision Natural language processing

	Operations research Parallel and distributed computing		Parallel high-performance computing: tools, algorithms, and software
Security and privacy		Security and privacy Biometric systems	Computer forensics Mathematics for cryptography Encoding and encryption Network security Automated software verification Secure systems design
Computer games		Game design and development Computer graphics	Game engines and interactive experience Methods for artificial intelligence Linguaggi di programmazione II Multimedia information systems
Complex Software Systems	Advanced databases (DB Technologies) Advanced databases (Mod. NoSQL) Logic for computer science	Software project management and evolution Automated software verification	Algorithm design Linguaggi di programmazione II Software testing Secure systems design

Artificial Intelligence

Referente: Prof. Piero Andrea Bonatti

Il corso di laurea magistrale in Informatica fornisce, mediante i suoi esami obbligatori, una estesa preparazione in Intelligenza Artificiale, relativamente ai metodi e alle tecniche oggi più diffusi. In particolare, i due moduli di *Machine Learning* coprono le tecniche di apprendimento automatico basate su statistica e reti neurali, incluso il *deep learning*. I corsi di *Intelligent web* e *Logic for computer science* forniscono le competenze necessarie per la rappresentazione della conoscenza e il ragionamento automatico, inclusi gli standard RDF/RDFS e W3C OWL2 per l'encoding di ontologie e annotazioni semantiche delle risorse web. Gli insegnamenti obbligatori di indirizzo sono *Methods for artificial intelligence* – che copre ulteriori aspetti dell'Intelligenza Artificiale classica, tra i quali l'inferenza probabilistica e gli algoritmi di ricerca – e *Computer vision*, che si occupa del riconoscimento e dell'interpretazione delle immagini. Lo studente può inoltre approfondire diversi aspetti specialistici dell'AI. Gli argomenti di robotica cognitiva (ovvero metodi e algoritmi per realizzare agenti software e robot intelligenti) si trovano nei corsi di *Intelligent robotics* – dove si approfondiscono, ad esempio, gli aspetti di pianificazione delle azioni, anche in sistemi multirobot e team misti persone-robot – e *Human-robot interaction*, dove sono di particolare importanza l'analisi e l'interpretazione del comportamento degli utenti umani. L'elaborazione del linguaggio naturale (italiano, inglese, ecc.) è trattata in *Natural language processing*. Il corso di *Formal Methods for Strategic Reasoning* fornisce metodi teorici che trovano applicazione concreta in diversi campi, dai giochi alle aste on-line. Infine, gli aspetti etici e l'impatto sociale dell'Intelligenza Artificiale vengono trattati nel corso *Social, ethical, and psychological issues in artificial intelligence*.

Insegnamenti obbligatori

- Intelligent web
- Logic for computer science
- Machine learning (Mod. Statistical learning, Mod. Neural networks and deep learning)

Insegnamenti a scelta vincolata:

- Computer vision
- Methods for artificial intelligence

Insegnamenti consigliati - tematica robotica cognitiva

- Human-robot interaction
- Intelligent robotics

Altri insegnamenti consigliati su temi specialistici

- Natural language processing
- Social, ethical, and psychological issues in artificial intelligence
- Formal methods for strategic reasoning

Data Science

Referenti: Proff. Anna Corazza, Sergio Di Martino, Francesco Isgro

La diffusione della Data Science in campi tra loro anche molto diversi, quali la bioinformatica, la medicina, le scienze sociali, gli studi di marketing e molti altri ancora, ha moltiplicato le richieste di esperti in queste tecniche, i cosiddetti *data scientists*, le cui competenze si devono basare su una solida base informatica per la gestione di grosse moli di dati, il machine learning e la statistica avanzata. Tenendo in considerazione l'obbligatorietà dei due moduli di Advanced databases (DB Technologies, NoSQL), dei due moduli di Machine Learning (Statistical Learning e Neural Networks and Deep Learning), e del corso di Parallel and distributed computing, il piano di studi risultante prevede l'inserimento di due insegnamenti obbligatori di percorso, il primo volto agli aspetti statistici dell'analisi (Data analytics), il secondo alla progettazione di algoritmi sofisticati (Algorithm design). Al completamento del curriculum si può scegliere di approfondire le metodologie per l'analisi (Ottimizzazione Combinatoria, Calcolo parallelo e distribuito mod. B) oppure un particolare dominio di indagine, quali Bioinformatics, Natural Language Processing, Computer Vision.

Insegnamenti obbligatori

- Advanced databases (Mod. DB Technologies, Mod. NoSQL)
- Machine learning (Mod. Statistical learning, Mod. Neural networks and deep learning)

Insegnamenti a scelta vincolata:

- Data analytics
- Algorithm design

Insegnamenti consigliati - metodologici

- Combinatorial optimization
- Biometric Systems

Insegnamenti consigliati - domini di indagine

- Bioinformatics
- Natural Language Processing
- Computer Vision

Security and privacy

Referente: Prof. Piero Andrea Bonatti

Il Piano di Studi orientato alla sicurezza prevede un corso introduttivo a largo spettro sull'argomento e un corso di introduzione alle tecniche di autenticazione biometrica e di analisi automatica dei video di sicurezza. Per approfondire ulteriormente gli argomenti di sicurezza, lo studente può selezionare diversi insegnamenti che ne coprono tutti gli aspetti principali. Chi predilige tematiche di ingegneria del software può seguire *Secure systems design*, che illustra i temi di security e privacy *by design*. Il legame con gli aspetti legali e processuali e le tecniche di investigazione sono illustrate in *Computer Forensics*. La sicurezza delle reti informatiche è trattata in *Network security*. I metodi formali per garantire l'affidabilità del software (trusted computing bases) sono introdotti in *Automated software verification*. Crittografia, codici e le basi algebriche dei moderni crittosistemi si trovano in *Encoding and encryption* e *Mathematics for cryptography*. Gli ulteriori esami a scelta consigliati completano la preparazione tecnologica e metodologica di più comune utilizzo.

Insegnamenti a scelta vincolata

- Security and privacy
- Biometric systems

Insegnamenti consigliati - tematica sicurezza

- Automated software verification
- Computer Forensics
- Encoding and encryption
- Mathematics for cryptography
- Network security
- Secure systems design

Insegnamenti consigliati - tematica applicazioni su rete

- Linguaggi di programmazione II o Tecnologie web

Computer games

Referente: Prof. Marco Faella

Il percorso in Computer Games offre gli strumenti tecnici e teorici fondamentali per affrontare la progettazione e lo sviluppo di videogiochi ed altri sistemi software complessi. Infatti, oltre al loro valore commerciale, educativo e di intrattenimento, i videogiochi rappresentano un caso notevole nell'ambito dei sistemi software, essendo caratterizzati dalla stretta interazione tra molteplici sottosistemi concorrenti, in presenza di stringenti requisiti di performance.

I corsi obbligatori e quelli consigliati di questo percorso approfondiscono da un lato delle tecniche generali di programmazione avanzata, e dall'altro le principali funzionalità utili ai sistemi di intrattenimento digitale, come l'intelligenza artificiale e la grafica tridimensionale. In aggiunta, vengono illustrati gli elementi di base per la progettazione dell'esperienza di gioco (game design).

Insegnamenti a scelta vincolata

- Game design and development
- Computer graphics

Insegnamenti consigliati

- Game design and interactive experience
- Methods for artificial intelligence
- Linguaggi di programmazione II
- Multimedia Information Systems

Complex Software Systems

Referenti: Prof. Sergio Di Martino, Massimo Benerecetti

Il Piano di Studi è orientato ad approfondire le tecniche e le metodologie per lo sviluppo dei sistemi informatici complessi e di alta qualità, con particolare attenzione alle fasi di progettazione, sviluppo, verifica e validazione del software. Il Piano prevede un primo insegnamento a scelta vincolata (Software Project Management and Evolution) in cui saranno presentate le più moderne metodologie dell'Ingegneria del Software per gestire progetti complessi, finalizzate a sviluppare prodotti di alta qualità in maniera efficace, anche in scenari di Manutenzione del Software, con Architetture distribuite ed eterogenee. Il secondo insegnamento a scelta vincolata (Automated Software Verification) pone particolare enfasi sulle metodologie per la specifica e la verifica di proprietà descritte e attese dei sistemi informatici, incluse le tecniche di tipo automatico o automatizzabile (in particolare il model checking), che sono alla base delle certificazioni di software che operano in contesti critici, quali ad esempio l'ambito medico, automotive o aeronautico. Ulteriori esami consigliati dal piano di studi sono tesi a completare il quadro tecnico-metodologico.

Insegnamenti obbligatori

- Advanced databases (Mod. DB Technologies, Mod. NoSQL)
- Logic for computer science

Insegnamenti a scelta vincolata

- Software Project Management and Evolution
- Automated Software Verification

Altri insegnamenti consigliati

- Algorithm Design
- Linguaggi di programmazione II
- Software Testing
- Secure Systems Design

Programma Erasmus

Erasmus+ è il programma di mobilità voluto e finanziato dall'Unione Europea che consente anche agli studenti universitari di trascorrere un periodo di studio presso una Università straniera convenzionata, con un contributo finanziario UE e con la possibilità di seguire corsi, sostenere esami e di usufruire delle strutture disponibili senza pagare a questa le tasse di iscrizione.

L'attività didattica (esami, tirocini) svolta in Erasmus+ viene poi riconosciuta, sia in termini di crediti che di voti, dall'Università di appartenenza.

In formazioni più dettagliate possono essere recuperate nel sito del Corso di Studi all'indirizzo:

<https://informatica.dieti.unina.it/index.php/it/servizi-per-gli-studenti/erasmus>

Il referente del Programma Erasmus del Corso di Studi è il Prof. Luigi Sauro.

Esame finale e attribuzione del punteggio finale

L'esame finale di laurea consiste nella redazione di un elaborato di tesi e nella presentazione dello stesso alla Commissione di Laurea.

L'elaborato di tesi è un documento redatto dal candidato sotto la guida del Relatore. Non sono previsti formati specifici per gli elaborati di tesi.

La presentazione dell'elaborato di tesi ha una durata massima di 20 minuti. Durante la presentazione, gli studenti possono avvalersi di supporti audio-visivi (e.g.: presentazione realizzata con Microsoft PowerPoint o strumenti simili). Al termine della presentazione, i membri della Commissione di Laurea possono porre domande ai candidati sul lavoro svolto.

Al termine dell'esame viene attribuito un punteggio complessivo ottenuto dalla somma dei seguenti addendi:

1. Il voto medio degli esami del candidato, espresso in centodecimali;
2. Il punteggio espresso dalla Commissione di Laurea in seguito alla prova del candidato, compreso tra 1 e 8 punti;

Qualora il voto medio degli esami del candidato, espresso in trentesimi, sia maggiore o uguale di 28 e la valutazione finale sia massima (110/110), la Commissione di Laurea può concedere la lode, con decisione unanime.

La richiesta di lode va comunicata dal relatore alla Commissione di Laurea almeno una settimana prima della data dell'esame finale, ove si ritenga che il candidato ne possieda i requisiti.

Calendario Attività Didattiche

Calendario delle attività didattiche - a.a. 2025/2026

Il calendario delle attività didattiche è stabilito dalla Scuola Politecnica e delle Scienze di Base ed è consultabile alla [pagina dedicata sul sito web della Scuola](#).

Per l'a.a. 2025/2026, le lezioni del primo periodo didattico (primo semestre) si terranno orientativamente **dal 15 settembre al 19 dicembre 2025**. Le lezioni del secondo periodo didattico (secondo semestre) si terranno orientativamente **dal 2 marzo al 12 giugno 2026**.

Le date precise saranno pubblicate non appena disponibili.

Nei periodi di esami, sono garantiti cinque appelli ordinari e due appelli di recupero (come definiti nel Regolamento Didattico di Ateneo), definiti come segue:

- Appelli ordinari obbligatori: un appello nel mese di gennaio 2026, un appello nel mese di febbraio 2026, un appello nel mese di giugno 2026, un appello nel mese di luglio 2026 ed un appello nel mese di settembre 2026.
- Appelli di recupero obbligatori: un appello nel mese di Marzo 2026, un appello nel periodo Ottobre-Novembre 2026.
 - **Le date precise saranno pubblicate non appena disponibili.**

Orario degli insegnamenti

L'orario degli insegnamenti è consultabile on-line all'indirizzo:

<https://informatica.dieti.unina.it/index.php/it/servizi-per-gli-studenti/orario-delle-lezioni/orario-laurea-magistrale>

Calendario degli Esami di profitto

Il calendario degli esami di profitto è consultabile on-line all'indirizzo:

<https://informatica.dieti.unina.it/index.php/it/servizi-per-gli-studenti/calendario-esami/calendario-esami-laurea-magistrale>

Calendario degli Esami di Laurea

Per gli appelli previsti nell'anno solare si consulti l'indirizzo:

<http://www.scuolapsb.unina.it/index.php/laurea-ingegneria>

Le commissioni degli esami di laurea sono consultabile on-line all'indirizzo:

<https://informatica.dieti.unina.it/index.php/it/servizi-per-gli-studenti/calendario-esami-di-laurea/laurea-magistrale>

Referenti del Corso di Studi

Coordinatore Didattico dei Corsi di Studio in Informatica:

Prof. Sergio Di Martino – Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - tel. 081/679272 - e-mail: cds.informatica@unina.it

Referente del Corso di Laurea per il Programma ERASMUS+:

Prof. Luigi Sauro – luigi.sauro@unina.it

Referente Orientamento

Prof. Daniel Riccio – daniel.riccio@unina.it

Responsabile dei Contenuti del Sito Web

Prof. Luigi Libero Lucio Starace – luigiliberolucio.starace@unina.it

Referente Pratiche Studenti

Prof.ssa Mara Sangiovanni – mara.sangiovanni@unina.it

Rappresentanti degli studenti:

Dott. Mario Barbato – Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - e-mail: mario.barbato5@studenti.unina.it

Contatti e Strutture

I corsi sono erogati nel plesso di [Via Claudio, 21, 80125 Napoli NA](#).

Siti web utili:

- Sito web del Corso di Studio: <https://informatica.dieti.unina.it/>
- Sito web del Dipartimento: <https://dieti.unina.it>
- Sito web della Scuola: <http://www.scuolapsb.unina.it>
- Sito web di Ateneo: www.unina.it
- Portale Orientamento: www.orientamento.unina.it

Canali Social ufficiali:

- Instagram: <https://www.instagram.com/unina.informatica/>
- LinkedIn: <https://www.linkedin.com/company/informatica-federico-ii/>
- Facebook: <https://www.facebook.com/InformaticaUnina/>
- Canale Telegram Ufficiale: https://t.me/informatica_unina

Schede Degli Insegnamenti

Di seguito vengono riportate le schede degli insegnamenti. Per ogni insegnamento è presente la versione italiana e la versione inglese della scheda. Sono, inoltre, indicati gli insegnamenti obbligatori per ogni percorso tematico, con un simbolo quadrato del colore corrispondente, con bordatura.



Gli insegnamenti a scelta vincolata per ogni percorso tematico sono indicati con un simbolo quadrato del colore corrispondente, senza bordatura.



SCHEDA DELL' INSEGNAMENTO DI Advanced Databases (DB Technologies)



TITOLO INSEGNAMENTO IN INGLESE Advanced Databases - DB technologies

Docente: Daniel Riccio

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X			X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

La finalità del modulo del corso è di introdurre gli aspetti tecnologici principali di un DBMS relazionale. In particolare, lo studente potrà apprendere i problemi e le tecniche utili alla progettazione fisica di una base di dati e agli aspetti più rilevanti per l'amministrazione di una base di dati di dimensioni medio-grandi. Il corso si concentrerà, tra le altre, sulle tecniche di indicizzazione, di riscrittura delle interrogazioni e sulla ottimizzazione delle interrogazioni. Le tecniche di riscrittura e ottimizzazione saranno applicate anche nell'ambito del Datawarehousing. Lo studente verrà indirizzato allo studio delle metodologie standard di progettazione e implementazione di un datawarehouse design (con un modello relazionale dei dati - ROLAP). Il modulo prevede anche una esperienza pratica di ottimizzazione di interrogazioni per una applicazione datawarehouse.

PROGRAMMA

Aspetti tecnologici di un database server: Transazioni, Controllo della concorrenza; Buffer management; Controllo di affidabilità. Progettazione fisica di un database. Strutture di indicizzazione: indici B-tree, indici B+-tree, indici bitmap, indici a liste invertite, indici per dati multidimensionali, R-tree indexes. Implementazione delle operazioni relazionali: proiezione, selezione, join, group by, ordinamento. Tecniche di ottimizzazione delle interrogazioni. Tecniche di riscrittura delle interrogazioni.

Introduzione al datawarehousing. Analisi e riconciliazione delle sorgenti dei dati. Modellazione e progettazione concettuale. ROLAP: modellazione e progettazione logica. Supporto del datawarehousing in Oracle: un caso di studio. Oracle: star queries; definizione delle dimensioni, rollup e cube GROUP BY, viste materializzate, gestione dei vincoli di consistenza, tecniche di riscrittura delle interrogazioni per viste materializzate.

MODALITA' DIDATTICHE

Lezioni in classe. Progetto di gruppo.
--

MATERIALE DIDATTICO

M. Golfarelli, S. Rizzi, Datawarehouse, McGrawHill.
Oracle database. Datawarehousing Guide. 11g Release 2 E25554-02.
P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone, Databases, McGraw-Hill
A. Albano, Costruire sistemi per basi di dati, Addison-Wesley.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta	X	Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	X
Altro	Progetto di gruppo: ottimizzazione di interrogazioni in un datawarehouse.					

TEACHING SUMMARY FOR THE COURSE Advanced Databases (DB Technologies)



Teacher: Daniel Riccio

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X			X		

Propaedeutic courses: None

OBJECTIVES

The aim of the course module is to introduce the technology of relational DBMS. In particular, the student is introduced to the main issues and technologies required for a competent design of the physical layer of a database and to face the administration issues of a database. In particular, the course will be focused on indexing techniques, query rewriting and optimization. Indexing and optimization techniques will be exploited then in the context of Datawarehousing. The student will be introduced to the standard methodologies of datawarehouse design and implementation (assuming a relational data model). The module also includes a practical experience of query optimization for a datawarehouse.

PROGRAM

Technology of a database server: Transactions, Concurrency control; Buffer management; Reliability control systems. Physical database design. Indexing structures: B-tree index, B+-tree index, bitmap index, inverted list index, indexes for multidimensional data, R-tree indexes. Implementation of the relational operations: projection, selection, join, group by, ordering. Query optimization techniques. Rewriting techniques.

Introduction to datawarehousing. Analysis and reconciliation of data sources. Conceptual modeling and design. ROLAP: logical modeling and design. Support for datawarehousing in Oracle: a case study. Oracle: star queries; definition of dimensions, rollup, and cube GROUP BY, materialized views, managing constraints, rewriting techniques for materialized views.

TEACHING MODALITIES

Lectures. Development of a group project.

TEACHING MATERIALS

M. Golfarelli, S. Rizzi, Datawarehouse, McGrawHill.

Oracle database. Datawarehousing Guide. 11g Release 2 E25554-02.

P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone, Databases, McGraw-Hill

A. Albano, Costruire sistemi per basi di dati, Addison-Wesley.

EXAM

The exam is given in form	Written and oral		Written only	X	Oral only	
In case of written exams, the tests are	Multiple choice		Open answer	X	Numeric tests	X
Other	Project: query optimization in the context of a datawarehouse.					

SCHEDA DELL' INSEGNAMENTO DI Advanced Databases (NoSQL)



TITOLO INSEGNAMENTO IN INGLESE Advanced Databases (NoSQL)

Docente: Andrea Cali

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

L'obiettivo del modulo è quello di presentare DBMS che adottano modelli dei dati alternativi a quello relazionale. Lo studio verrà indirizzato sia a modelli alternativi tradizionali quali il modello a oggetti ed il modello relazionale ad oggetti ed il modello dei dati semi-strutturato sia a considerare le proposte più recenti sviluppate nel contesto dei modelli NoSQL: DB colonnari, modello a grafo, modello key-value etc. Il fine complessivo del modulo è quello di orientare consapevolmente lo studente nella scelta dei modelli di dati e delle soluzioni tecnologiche più adeguate alla risoluzione dei problemi di memorizzazione e recupero dei dati.

PROGRAMMA

Limitazioni del modello relazionale dei dati. Basi di dati ad oggetti: definizione dei dati ODL e OQL. Basi di dati relazionali ad oggetti definizione dei dati ed interrogazioni. Il caso Object relational di Oracle. Basi di dati semi-strutturati. Definizione dei dati in XML (DDT e X-Schema). Interrogazione dei dati: XPATH e XQUERY. DBMS NoSQL: DBMS colonnari, orientati ai documenti, key-value, orientati ai grafi, orientati alle serie temporali. Rassegna dei DBMS NoSQL: MongoDB, Cassandra, InfluxDB, Neo4J etc. Database spaziali. Un caso di studio: Oracle Spatial.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Cattell R.G.C. , D. Berry et al, The Object Data Standard, Morgan Kaufmann Ed.
Oracle database. Object relational Developer's Guide. 12c Release 1 E53227-02.
Oracle database. Spatial Developer's Guide. 11g.
S. Abiteboul, I. Manolescu, P. Rigaux, M. Rousset, P. Senellart, Web Data Management, Cambridge University Press, webdam.inria.fr/Jorge/files/wdm.pdf

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta	X	Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	X
Altro						

TEACHING SUMMARY FOR THE COURSE Advanced Databases (NoSQL)



Teacher: Andrea Calì

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

The module is focused on DBMS adopting a data model which either extends the relational data model or adopt alternative data models. In particular, the module will discuss data definition and query languages for DBMS adopting the object model, object relational model, semi-structured model and the models following the recent NoSQL trend for databases: columnar DB, graph DB, document DB, key-value DB etc. The goal of the course is to allow the student to choose properly data models and DB technologies depending on the specific concrete needs of the problem under modelling and design.

PROGRAM

Limits of the relational data model. Object databases: data definition in ODL e query in OQL. Object relational databases; data definition and query. The Object relational model of Oracle. Semi-structured databases. Data definition in XML (DDT e X-Schema). Data query in XML: XPATH e XQUERY. DBMS NoSQL: column DBMS, document-oriented DBMS, key-value oriented DBMS, graph oriented DBMS, time series oriented DBMS. Overview of NoSQL DBMS: MongoDB, Cassandra, InfluxDB, Neo4J etc. Spatial database. A case studi: Oracle Spatial.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

Cattell R.G.C. , D. Berry et al, The Object Data Standard, Morgan Kaufmann Ed.
 Oracle database. Object relational Developer's Guide. 12c Release 1 E53227-02.
 Oracle database. Spatial Developer's Guide. 11g.
 S. Abiteboul, I. Manolescu, P. Rigaux, M. Rousset, P. Senellart, Web Data Managment, Cambridge University Press, webdam.inria.fr/Jorge/files/wdm.pdf

EXAM

The exam is given in form	Written and oral		Written only	X	Oral only	
In case of written exams, the tests are	Multiple choice		Open answer	X	Numeric	X
Other						

SCHEDA DELL' INSEGNAMENTO DI Algorithm Design



TITOLO INSEGNAMENTO IN INGLESE Algorithm Design

Docente: Fabio Mogavero

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso intende fornire un'introduzione alle tecniche avanzate di progettazione degli algoritmi, alla complessità computazionale e alla trattabilità dei problemi. Vengono, in particolare, presentate le principali tecniche di dimostrazione di correttezza, esaminate le tecniche di progettazione greedy e di programmazione dinamica, con applicazioni alla soluzione di vari problemi di ottimizzazione, di compressione dei dati e problemi su grafi pesati. Vengono introdotte le classi di complessità P e NP e il concetto di NP-completezza e di riduzione tra problemi. Vengono infine presentate tecniche di progettazione ed analisi di algoritmi approssimati e di algoritmi randomizzati.

PROGRAMMA

Il problema della correttezza degli algoritmi: dimostrazioni per induzione e dimostrazioni di correttezza di algoritmi ricorsivi. Tecniche di progettazione di algoritmi: tecnica divide et impera, algoritmi greedy e programmazione dinamica, applicati alla soluzione di problemi di ottimizzazione (problema dello zaino, percorsi minimi su grafi pesati, allineamento di sequenze, scheduling di attività). Algoritmi per il calcolo del flusso su reti e loro applicazioni: massimo flusso e segmentazione di immagini. Problemi trattabili e non trattabili: le principali classi di complessità (P e NP); riduzioni polinomiali tra problemi; il concetto di NP-completezza; esempi di problemi NP-completi; dimostrazioni di NP-completezza. Introduzione all'intrattabilità computazionale: algoritmi approssimati e fattore di approssimazione. Algoritmi di ricerca locale e applicazioni a problemi di machine learning.

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

Jon Kleinberg e Eva Tardos: Algorithm Design, Addison-Wesley

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY FOR THE COURSE Algorithm Design



Teacher: Fabio Mogavero

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X			X	

Propaedeutic courses: None

OBJECTIVES

The course is meant to provide an introduction to advanced algorithm design techniques, computational complexity and the notion of tractable problem. More specifically, the course covers the most common correctness proof techniques, and discusses the design techniques such as greedy algorithms and dynamic programming, with application to the solution of optimization problems (e.g., task scheduling, problems on weighted graphs, clustering, etc.). The complexity classes P and NP are also presented, as well as the notions of NP-completeness and problem reductions. Approximation techniques and local search algorithms for the solution of computationally difficult problems are also discussed.

PROGRAM

Algorithms correctness proofs: proofs by induction and the correctness of recursive algorithms. Algorithm design techniques: divide et impera, greedy algorithms and dynamic programming, with applications to the solution of optimization problems (e.g., knapsack problem, shortest paths in weighted graphs, sequence alignment, task scheduling). Network flow algorithms and their applications: maximum flow and image segmentation. Tractable and non-tractable problems: the complexity classes P and NP; polynomial reductions between problems; NP-completeness and examples of NP-complete problems. Tackling non-tractable problems: approximation algorithms and approximation factor; local search algorithms with applications to neural networks and classification problems.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

Jon Kleinberg e Eva Tardos: Algorithm Design, Addison-Wesley

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☒	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Automated Software Verification



TITOLO INSEGNAMENTO IN INGLESE Automated Software Verification

Docente: da definire

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso si propone di fornire le nozioni di base sottostanti il problema della verifica automatica di proprietà di correttezza di sistemi informatici software e hardware. In particolare, verranno introdotte e studiate le tecniche di Model Checking. I principali obiettivi del corso sono quelli di familiarizzare lo studente con gli strumenti fondamentali per la comprensione e l'utilizzo degli strumenti automatici di verifica, lo studio dei principali algoritmi di verifica automatica, alcune delle più importanti ottimizzazioni ed estensioni delle tecniche di Model Checking in uso nelle realtà produttive coinvolte nello sviluppo di sistemi sia hardware che software, sistemi embedded (ad es., sistemi automotive e internet-of-things) e sistemi safety-critical (ad es., sistemi di controllo di traffico ferroviario e aereo).

PROGRAMMA

La parte iniziale del corso riguarderà lo sviluppo dei prerequisiti relativi agli strumenti di base per la modellazione di sistemi software e hardware e delle loro proprietà di correttezza. In questa fase, verranno richiamati elementi di logica classica e logica modale; automi a stati finiti su parole finite e su parole infinite; tecniche di base e linguaggi per la modellazione di sistemi software e hardware tramite sistemi a transizione e macchine a stati. Verranno inoltre descritti e studiati linguaggi per la specifica di proprietà di correttezza, nello specifico logiche temporali lineari (LTL) e ramificate (CTL). In seguito, verranno descritte le tecniche algoritmiche per la soluzione automatica dei problemi di verifica: Model Checking esplicito per proprietà CTL; Model Checking per proprietà LTL basato su automi; il problema dell'esplosione dello spazio degli stati e il Model Checking simbolico basato su OBDD. Durante il corso verranno, inoltre, impiegati tool di model checking (NuSMV e SPIN) con esempi di utilizzo.

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

E. M. Clarke, O. Grumberg D. Peled: "Model Checking".

Lucidi utilizzati per le lezioni frontali del corso

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Automated Software Verification



Teacher: to be defined

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X			X	

Propaedeutical courses: None

OBJECTIVES

The course is intended to provide the basic notions underlying the software and hardware verification problem. The main focus will be on automatic verification approaches and more specifically, the students will be introduced to the classic Model Checking techniques. The goal is to familiarize students with the main principles and tools of automatic verification of software and hardware systems, that are typically used in the development of safety-critical systems (e.g., railways and flight control systems) and embedded systems (e.g., automotive applications and internet-of-things). To this end, modeling techniques, automatic verification algorithms, together with the most common optimization techniques, will be covered in the course.

PROGRAM

The first part of the course will cover the basics concerning the modeling of software and hardware systems and the specification of the correctness properties. These include classical logics, finite state automata on finite and infinite words; system modeling languages, based on transition systems and finite state machines; property specification languages, such as linear temporal logics (LTL) and branching time temporal logics (CTL). The rest of the course will cover algorithmic techniques for the solution of the verification problem: Model Checking algorithms for linear time properties based on automata; explicit and implicit Model Checking algorithms for branching time logics; the state space explosion problem and the symbolic approach based on ordered binary decision diagrams (OBDD). During the course, Model Checking tools, such as SPIN and NuSMV, will be presented and experimented with.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

E. M. Clarke, O. Grumberg D. Peled: "Model Checking".

Slides used in the lectures.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Bioinformatics

TITOLO INSEGNAMENTO IN INGLESE Bioinformatics

Docente: tace per il solo a.a. 2025/2026

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso ha l'obiettivo di introdurre lo studente ai problemi e ai metodi principali usati nella descrizione di fenomeni complessi della biologia molecolare e cellulare, mediante l'applicazione di approcci quantitativi della matematica, della statistica, dell'informatica e della fisica. Al termine del corso lo studente sarà capace di sviluppare, implementare e applicare algoritmi e concetti della biologia computazionale e della bioinformatica a problemi reali.

PROGRAMMA

Introduzione ai concetti di base di biologia molecolare; concetti di base di probabilità e statistica e algoritmi di bioinformatica. Tecnologie high-throughput e relative dati; tecniche di machine learning e inferenza di base. Simulazioni al Computer, Gibbs sampling e Ottimizzazioni (Monte Carlo, Simulated Annealing); Biological Networks. Applicazioni a problemi reali nelle scienze e nelle tecnologie biomediche.

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

Nello Cristianini, Introduction to computational genomics (Cambridge Univ. Press).
Note fornite dal docente.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Bioinformatics

Teacher: not active only for the academic year 2025/2026

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

Propaedeutic courses: None

OBJECTIVE

The course aims to introduce the student to the main issues and methods for the description of complex phenomena of molecular and cell biology through quantitative approaches from mathematics, statistics, informatics and physics. At the end of the course, the student will be able to design, implement and apply computational biology and bioinformatics algorithms and analysis concepts to a real problem.

PROGRAM

Introduction to Basic Molecular Biology; Basic Concepts in Probability, Statistics and Bioinformatics Algorithms.
High-throughput technologies and data; Basic Inference and Machine Learning Techniques.
Computer simulations, Gibbs sampling and Optimization (Monte Carlo, Simulated Annealing); Biological Networks
Real-world applications in science and biomedical technologies.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

Nello Cristianini, Introduction to computational genomics (Cambridge Univ. Press).
Notes provided by the lecturer.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Mutliple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Biometric Systems

TITOLO INSEGNAMENTO IN INGLESE Biometric Systems

Docente: Daniel Riccio

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso ha come obiettivo quello di fornire allo studente una panoramica dei sistemi biometrici nella accezione più ampia del termine, ovvero nella sua duplice interpretazione di sistemi per l'analisi di caratteristiche fisiologiche ai fini di garantire la sicurezza fisica/logica di luoghi e persone e di supportare la diagnosi medica.

Nella sua prima parte il corso si concentra sull'analisi di caratteristiche handcrafted e deep features estratte da impronta, volto, iride e vene, per poi spostarsi sulla loro classificazione e sugli schemi di fusione. La seconda parte del corso è orientata allo studio di caratteristiche biometriche estratte da immagini 2D e volumi di slice 3D allo scopo di supportare la diagnosi del medico in ambito clinico.

Al termine del corso lo studente sarà capace di progettare, implementare e applicare algoritmi per il trattamento di caratteristiche biometriche nella soluzione di problemi reali.

PROGRAMMA

I sensori ed i segnali in ambito biometrico. Le caratteristiche handcrafted 1D, 2D e 3D. Le Deep features.

I tratti biometrici:

- impronta;
- volto;
- iride;
- vene.

Confronto di chiavi biometriche e indici di similarità: distanze e classificatori. Schemi di selezione e fusione. Applicazioni biometriche per la sicurezza. Analisi di segnali in ambito medico 1D (EEG, ECG). Analisi di segnali 2D (X-ray films, MRI, CT). Analisi di immagini al microscopio elettronico (IF, WSI). Analisi di segnali 3D (MRI, CT, video). I sistemi CAD in ambito medico. Applicazioni biometriche per la medicina.

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

Yingzi (Eliza) Du. <i>Biometrics: From Fiction to Practice</i> . CRC Press, 2013.
Najarian, Kayvan, and Robert Splinter. <i>Biomedical signal and image processing</i> . Second Edition. CRC press, 2012.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Biometric Systems



Teacher: Daniel Riccio

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

The main goal is to provide the student with an overview of biometrics. The course takes the word <<biometrics>> as having the dual meaning of exploiting physiological/behavioral traits for people authentication and measuring physiological parameters to support a medical diagnosis.

The first part of the course focuses on most important biometric traits, such as fingerprints, face, iris and veins. The main topics are the extraction of handcrafted and deep features, the template matching and classification and the multi-modal/multi-biometric fusion.

The second part of the course concentrates on the analysis of biometric features extracted from 1D signals, 2D images and 3D slice volumes to support the diagnosis of physicians in a clinical setting.

At the end of the course, the student will be able to design, implement and apply algorithms to exploit biometric features in the solution of real problems.

PROGRAM

Biometric sensors and signals. Handcrafted features 1D, 2D and 3D. Deep features.

Biometric traits:

- footprint;
- face;
- iris;
- veins.

Template matching and score: distances and classifiers. Feature selection and fusion schemes. Biometric applications for security. Analysis of 1D physiological signals (EEG, ECG). Analysis of 2D physiological signals (X-ray films, MRI slices, CT slices). Histopathological image analysis (IF, WSI). Analysis of 3D physiological signals (MRI, CT, video). CAD systems in the clinical setting. Biometric applications for medicine.

TEACHING MODALITIES

Lectures.

TEACHING MATERIAL

Yingzi (Eliza) Du. <i>Biometrics: From Fiction to Practice</i> . CRC Press, 2013.
Najarian, Kayvan, and Robert Splinter. <i>Biomedical signal and image processing</i> . Second Edition. CRC press, 2012.

EXAM

The exam is given in form	Written and oral		Written only		Oral only	X
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Combinatorial Optimization

TITOLO INSEGNAMENTO IN INGLESE Combinatorial Optimization

Docente: Paola Festa

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
MATH-06/A	6		X			X	X

Insegnamenti propedeutici previsti: Ricerca Operativa

OBIETTIVI FORMATIVI

Questo insegnamento si prefigge quale obiettivo principale l'introduzione degli studenti all'uso dei modelli di programmazione matematica con particolare attenzione rivolta ai modelli di ottimizzazione a variabili intere. Allo studio teorico di questi problemi viene affiancata la descrizione delle loro applicazioni in scenari del mondo reale, inclusi il controllo ottimo, le comunicazioni, la logistica, i servizi e la produzione industriale. Per quanto riguarda i modelli di programmazione a variabili intere con regione ammissibile finita (problemi combinatorici sia lineari che non lineari), il corso mira a fornire un trattamento completo e rigoroso della loro classificazione computazionale. Per quei problemi computazionalmente intrattabili, oltre ai metodi di soluzione esatti, il corso si prefigge di illustrare anche metodi più sofisticati, come algoritmi di approssimazione e algoritmi euristici e metaeuristici.

PROGRAMMA

Introduzione all'Ottimizzazione Combinatoria: definizioni di problema di ottimizzazione e di problemi di decisione; definizione di un generico problema di ottimizzazione combinatoria lineare; definizione di un generico problema di ottimizzazione combinatoria non lineare; proprietà della regione ammissibile di un problema di ottimizzazione combinatoria. Funzioni di Karp-riducibilità polinomialmente calcolabili. Classi di Complessità Computazionale: classe P e sue proprietà; classe NP e sue proprietà; classe NP-ardua e sue proprietà; classe NP-completa e sue proprietà; classe fortemente NP-completa e sue proprietà. Classificazione dei metodi di soluzione: metodi esatti; metodi approssimazione; metodi euristici. Metodi esatti: Upper e Lower Bounds; Branch & Bound; Branch & Cut; Piani di Taglio; Programmazione Dinamica. Fondamenti teorici per i metodi greedy – Teoria delle Matroidi: definizione di matroide e sue proprietà; algoritmi greedy per problemi su matroidi pesate e dimostrazione delle loro correttezza; esempi di problema di ottimizzazione combinatoria risolvibili all'ottimo tramite tecnica greedy in quanto riconducibili a problemi su matroidi pesate: il Problema dell'Albero di Copertura Minimo di un grafo; un Problema di Schedulazione. Algoritmi di Approssimazione: definizione e calcolo del rapporto di prestazione nel caso peggiore (errore assoluto) di un algoritmo di approssimazione; definizione di errore relativo nel caso peggiore di un algoritmo di approssimazione. Classi di Approssimazione: classe APX e sue proprietà; classe PTAS e sue proprietà; classe FPTAS e sue proprietà. Problema del Vertex Cover Minimo di un grafo: descrizione verbale e logico-matematica; teorema dell'approssimabilità; un algoritmo di approssimazione di tipo greedy: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; un algoritmo di approssimazione di tipo random: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale. Problema del Massimo Insieme Indipendente di un grafo: descrizione verbale e logico-matematica; teorema dell'inapprossimabilità; un algoritmo di tipo greedy non di approssimazione: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale. Classificazione dei Metodi Euristici. Definizione di Neighborhood di una soluzione. Procedure di Ricerca Locale. Algoritmi Metaeuristici: algoritmi genetici; algoritmi tabù search; algoritmi simulated annealing; algoritmi GRASP (Greedy Randomized Adaptive Search Procedure): un algoritmo GRASP per il Problema Max-Cut. Il Problema dello Zaino 0/1: descrizione verbale e logico-matematica; rilassamento continuo: il Problema dello Zaino Frazionario;

un algoritmo Branch & Bound; due algoritmi di Programmazione Dinamica; due upper bounds; un algoritmo 1/2 approssimato: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; un PTAS: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; un FPTAS: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; Il Problema del Commesso Viaggiatore (TSP): descrizione verbale e logico-matematica; teorema dell'inapprossimabilità; un algoritmo Branch & Bound; varianti del Problema dello Commesso Viaggiatore (TSP): TSP su grafi generici; TSP grafico; TSP asimmetrico; TSP bottleneck. un algoritmo 2-approssimato per il TSP simmetrico con disuguaglianza triangolare: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; un algoritmo 3/2-approssimato per il TSP simmetrico con disuguaglianza triangolare: dimostrazione della convergenza; calcolo dell'errore; analisi della complessità computazionale; algoritmi euristici per il TSP standard; algoritmi di costruzione di una soluzione ammissibile per il TSP standard; insiemi neighborhoods di una soluzione; algoritmi di ricerca locale: 2-opt exchange; k-opt exchange; analisi della complessità computazionale di 2-k-opt exchange. Metaeuristiche per il TSP standard: un algoritmo genetico; un algoritmo tabù search; un algoritmo simulated annealing; un algoritmo GRASP.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Appunti e Dispense del Corso

P. Festa. Ottimizzazione Combinatoria, UTET Università – De Agostini Scuola, 2017.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY FOR THE COURSE Combinatorial Optimization

Teacher: Paola Festa

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
MATH-06/A	6		X			X	X

Propaedeutic courses: Operations research

OBJECTIVES

This course has as main objective the introduction of students to the use of mathematical programming models with particular attention paid to integer variable optimization models. The theoretical study of these problems is accompanied by the description of their applications in real world scenarios, including optimal control, communications, logistics, services and industrial production.

With regard to integer optimization problems with finite feasible region (both linear and nonlinear combinatorial problems), the course aims to provide a complete and rigorous treatment of their computational classification. For those computationally intractable problems, in addition to the exact solution methods, the course also aims to illustrate more sophisticated methods, such as approximation algorithms and heuristic and metaheuristic algorithms.

PROGRAM

Introduction to Combinatorial Optimization: definitions of optimization problem and decision problems; definition of a generic problem of linear combinatorial optimization; definition of a generic problem of nonlinear combinatorial optimization; properties of the feasible region of a combinatorial optimization problem. Polynomially calculable Karp-reducibility functions. Computational Complexity Classes: class P and its properties; NP class and its properties; NP-arduous class and its properties; NP-complete class and its properties; strongly NP-complete class and its properties. Classification of solution methods: exact methods; approximation methods; heuristic methods. Exact methods: Upper and Lower Bounds; Branch & Bound; Branch & Cut; Cutting plans; Dynamic programming. Theoretical foundations for greedy methods - Matroid theory: definition of matroid and its properties; greedy algorithms for problems on weighed matroids and demonstration of their correctness; examples of combinatorial optimization problem that can be solved in the best way by greedy technique as they can be traced back to problems on weighted matrixes: the Minimum Coverage Tree Problem of a graph; a Scheduling Problem. Approximation algorithms: definition and calculation of the worst-case performance ratio (absolute error) of an approximation algorithm; definition of relative error in the worst case of an approximation algorithm. Approximation classes: APX class and its properties; PTAS class and its properties; FPTAS class and its properties. Vertex Minimum Cover problem of a graph: verbal and logical-mathematical description; approximability theorem; a greedy approximation algorithm: demonstration of convergence; error calculation; computational complexity analysis; a random approximation algorithm: demonstration of convergence; error calculation; computational complexity analysis. Problem of the Maximum Independent Set of a graph: verbal and logical-mathematical description; the inaccuracy theorem; a non-approximation greedy algorithm: demonstration of convergence; error calculation; computational complexity analysis. Classification of Heuristic Methods. Neighborhood definition of a solution. Local Search Procedures. Metaheuristic algorithms: genetic algorithms; taboo search algorithms; simulated annealing algorithms; GRASP algorithms (Greedy Randomized Adaptive Search Procedure): a GRASP algorithm for the Max-Cut Problem. The Backpack Problem 0/1: verbal and logical-mathematical description; continuous relaxation: the Fractional Backpack Problem; a Branch & Bound algorithm; two Dynamic Programming algorithms; two upper bounds; an approximate 1/2 algorithm: demonstration of convergence; error calculation; computational complexity analysis; a PTAS: demonstration of convergence; error calculation; computational complexity analysis; an FPTAS: demonstration of convergence; error calculation; computational complexity

analysis; The Traveling Salesman Problem (TSP): verbal and logical-mathematical description; the inaccuracy theorem; a Branch & Bound algorithm; variants of the Traveling Salesman Problem (TSP): TSP on generic graphs; Graphic TSP; Asymmetric TSP; TSP bottleneck. a 2-approximate algorithm for symmetric TSP with triangular inequality: demonstration of convergence; error calculation; computational complexity analysis; a 3/2-approximate algorithm for symmetric TSP with triangular inequality: demonstration of convergence; error calculation; computational complexity analysis; heuristic algorithms for the standard TSP; algorithms for building an admissible solution for the standard TSP; neighborhoods sets of a solution; local search algorithms: 2-opt exchange; k-opt exchange; computational complexity analysis of 2-k-opt exchange. Metaheuristics for the standard TSP: a genetic algorithm; a taboo search algorithm; a simulated annealing algorithm; a GRASP algorithm.

TEACHING MODALITIES

Frontal lessons. Exercises

TEACHING MATERIAL

Notes provided by the lecturer.

P. Festa. Ottimizzazione Combinatoria, UTET Università – De Agostini Scuola, 2017.

EXAMS

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exam, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Computational complexity

TITOLO INSEGNAMENTO IN INGLESE Computational complexity

Docente: Piero Andrea Bonatti

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Questo corso estende e completa la preparazione algoritmica spostando l'attenzione dalla complessità dei singoli algoritmi alla complessità intrinseca dei problemi, espressa rispetto alle risorse computazionali necessarie per la loro risoluzione. In questo modo si apprendono criteri per stimare l'ottimalità degli algoritmi, si identificano tecniche comuni per la risoluzione sistematica di ampie classi di problemi, si acquisiscono approcci più scientifici alla risoluzione di nuovi problemi. Verranno spiegate le relazioni tra consumo di memoria e lunghezza della computazione e il ruolo del non-determinismo nell'analisi dei problemi la cui complessità esatta non è nota. Verranno altresì analizzate le relazioni tra questi aspetti e aree applicative importanti quali crittografia, ricerca operativa e ottimizzazione combinatoria, quantum computing.

PROGRAMMA

Problemi e algoritmi: formulazioni intuitive e formalizzazioni mediante linguaggi e macchine di Turing multinastro. Misure appropriate di complessità in termini di spazio e di tempo. Teoremi di speedup. Confronti con altre formalizzazioni delle computazioni e tesi di Church (cenni). Classi di complessità, teoremi di gerarchia e teorema di Savitch. Riduzioni e completezza come formalizzazioni delle difficoltà relative e della complessità caratteristica dei problemi. Alcuni risultati di separazione. Problemi logici, su grafi e su insiemi completi per NP e coNP. Teorema di Cook. La gerarchia polinomiale e PSPACE. Relazioni con la crittografia moderna. Breve analisi dell'impatto del quantum computing sulla risoluzione di problemi e sulla crittografia. Cenni delle classi oltre PSPACE: problemi che richiedono risorse esponenziali e problemi indecidibili.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Christos H. Papadimitriou, *Computational complexity*. Addison-Wesley 1994, ISBN 978-0-201-53082-7, pp. I-XV, 1-523

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla	X	A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Computational complexity

Teacher: Piero Andrea Bonatti

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Prerequisites: None

OBJECTIVES

This course extends and completes the courses on algorithms by focussing on the intrinsic complexity of problems, as opposed to the complexity of individual algorithmic solutions. Complexity is expressed in terms of the computational resources needed for solving the given problem. This approach provides criteria for assessing algorithm optimality, identifying common and systematic problem-solving techniques for wide classes of problems, and it teaches a more scientific approach at the solution of new problems. Among the course's topics, we mention the relationships between memory requirements and computation time, and the role of nondeterminism in the analysis of problems whose exact complexity is not yet known. These theoretical tools will be applied in important application domains such as cryptography, operation research, combinatorial optimization, and quantum computing.

PROGRAM

Problems and algorithms: intuitive formulations and formalizations through languages and multi-tape Turing machines. Appropriate complexity measures based on time and space. Speedup theorems. Comparisons with other formalizations and Church's thesis. Complexity classes, hierarchy theorems and Savitch's theorem. Reductions and completeness as a formalization of relative hardness and intrinsic problem complexity. Some separation results. Complete problems for NP and coNP. Cook's theorem. The polynomial hierarchy and PSPACE. Relationships with modern cryptography. Brief analysis of the potential of quantum computing on efficient problem solving and cryptography. Classes beyond PSPACE: exponential resources and undecidable problems.

TEACHING MODALITIES

Front lessons, exercises.

TEACHING MATERIAL

Christos H. Papadimitriou, *Computational complexity*. Addison-Wesley 1994, ISBN 978-0-201-53082-7, pp. I-XV, 1-523

EXAM

The exam is given in form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice	X	Open answers	X	Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Computer Graphics



TITOLO INSEGNAMENTO IN INGLESE Computer graphics

Docente: da definire

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Fornire agli studenti i metodi di base e i fondamenti della progettazione degli algoritmi per la sintesi di immagini bi-tridimensionali tramite calcolatore. Fornire agli studenti alcune tecniche implementative e le conoscenze basilari degli strumenti software per la modellazione e la visualizzazione di oggetti tramite calcolatore.

PROGRAMMA

Introduzione a sistemi grafici, tecniche di composizione della scena. Pipeline grafica per illuminazione locale: dal modello al rendering dell'immagine finale. Metodi, algoritmi e software per proiezioni geometriche, illuminazione, ombreggiatura e tessitura delle superfici tridimensionali, rimozione delle superfici nascoste, ritaglio degli oggetti al di fuori dell'inquadratura (libreria OpenGL). Strumenti per la gestione di una interfaccia real-time (libreria glut). Metodi, algoritmi e software per la modellazione e la manipolazione di oggetti geometrici, rappresentazioni poligonali, curve, superfici di Bezier e NURBS (libreria GLU). Metodi, algoritmi e software per l'illuminazione globale. Programmazione dei processori grafici (GPU). Studio e sviluppo di un case study che riprenda concetti, algoritmi e software discussi, ed utilizzati in particolari ambiti applicativi.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

--

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Computer graphics



Teacher: to be defined

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X		X		

Propaedeutic courses: None.

OBJECTIVES

This course introduces students to the overall computing concepts behind creating synthetic images. It focuses on methods, algorithms and software needed to render 2D and 3D images.

PROGRAM

Introduction to Graphics pipeline and Graphics Hardware.

Methods, algorithms and software underlying geometric transformations including triangles, normals, interpolation, texture mapping, homogeneous coordinates and perspective.

Methods, algorithms and software underlying shading and clipping.

Lighting Equation: Local and Global Illumination, Radiosity, Ray Tracing, Bidirectional Ray Tracing, Light Maps.

Methods, algorithms and software underlying modeling curves and surfaces. Theory of parametric and implicit curve and surface models: polar forms, Bezier arcs and de Casteljau subdivision, continuity constraints, B-splines, tensor product, and triangular patch surfaces. Subdivision surfaces and multiresolution representations of geometry. Representations of solids and conversions among them. NURBS.

Programming models on GPU architectures.

The final project consists of developing a case study using concepts, algorithms and software addressed during the course.

TEACHING MODALITIES

Frontal lectures. Exercises.

TEACHING MATERIALS

--

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Computer Vision



TITOLO INSEGNAMENTO IN INGLESE Computer Vision

Docente: Francesco Isgrò

SSD	CFU	Anno di corso (I, II o III)			Semestre (I o II)		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso si propone di presentare allo studente le principali problematiche e metodi per l'elaborazione e l'interpretazione delle immagini digitali, nonché di fornire una descrizione operativa di alcuni dei modelli computazionali più significativi della visione 2D e 3D. Alla fine del corso, lo studente sarà in grado di progettare, implementare e applicare algoritmi di immagine a un problema reale.

PROGRAMMA

Introduzione alle tecniche di elaborazione delle immagini con particolare riguardo alle tecniche di miglioramento delle immagini nel dominio spaziale.

Estrazione di funzioni dalle immagini: contorni, curve parametriche (trasformazione di Hough) e punti caratteristici. Introduzione alla rappresentazione spazio-scala. HOG. SIFT: SURF.

Reti neurali convolutive: strati e funzioni di loss. Inizializzazione del peso. Regolarizzazione. Architetture per le CNN. Applicazioni delle CNN.

Visione stereo e ricostruzione 3D. Introduzione a coordinate omogenee. Calibrazione di una telecamera. Geometria di un sistema stereo. Ricostruzione 3D.

MODALITA' DIDATTICHE

Lezioni. Sviluppo di un progetto di gruppo.

MATERIALE DIDATTICO

M. Nixon, A. Aguado, Feature extraction and image processing, Academic Press

R. Szeliski, Computer Vision: algorithm and applications, Springer

Materiale fornito dal docente.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Computer vision



Teacher: Francesco Isgrò

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X		X		

Propaedeutic courses: None

OBJECTIVES

The course aims to introduce the student to the main issues and methods for the processing and interpretation of digital images, as well as to provide an operational description of some of the most significant computational models of 2D and 3D vision. At the end of the course, the student will be able to design, implement and apply image algorithms to a real problem.

PROGRAM

Introduction to Image Processing techniques with particular regard to Image Enhancement techniques in the spatial domain. Extraction of features from images: contours, parametric curves (Hough transform) and characteristic points. Introduction to the scale-space representation. HOG. SIFT: SURF.

Convolutional Neural Networks: layers and loss functions. Weight initialisation. Regularization. CNN architectures. Applications of CNN.

Stereo vision and 3D reconstruction. Introduction to homogeneous coordinates. Calibration of a camera. Geometry of a stereo system. 3D reconstruction.

TEACHING MODALITIES

Development of a group project.

TEACHING MATERIALS

M. Nixon, A. Aguado, Feature extraction and image processing, Academic Press

R. Szeliski, Computer Vision: algorithm and applications, Springer

Notes provided by the lecturer.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Computer Forensics

TITOLO INSEGNAMENTO IN INGLESE Computer Forensics

Docente: Lorenzo Laurato

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Prerequisiti: Sistemi Operativi, Reti di calcolatori I

OBIETTIVI FORMATIVI

Il corso si pone l'obiettivo di far acquisire agli studenti le competenze di base nell'ambito della Computer Forensics su aspetti teorici, tecnici, metodologie e regole giuridiche alle quali deve attenersi chi opera nel settore, con illustrazione delle tecniche paradigmatiche di indagine scientifica laddove è possibile ricorrere a prove in formato digitale sia per i casi di reati strettamente informatici, sia per gli altri tipi di illeciti in cui il dato informatico può rappresentare una prova, e relativa declinazione nel contesto normativo italiano.

PROGRAMMA

Introduzione all'informatica forense. Elementi, ruolo, criticità e approccio metodologico dell'informatica forense. Normative e aspetti pragmatici relativi alla costruzione della prova. Aspetti legali e tecnologici relativi all'attendibilità del dato informatico e al trattamento del reperto informatico - nello specifico la disk forensics, e il trattamento dei file systems per la corretta acquisizione e la ricostruzione di informazioni. Strumenti Hardware e Software utilizzati nella digital forensics, ivi compresi la network forensics, la mobile forensics e l'embedded forensics. Metodologie per l'acquisizione di dati crittografati.

MODALITA' DIDATTICHE

Lezioni frontali ed esercitazioni.

MATERIALE DIDATTICO

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla	X	A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Computer Forensics

Teacher: Lorenzo Laurato

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

Prerequisites: Operating Systems, Computer Networks I

OBJECTIVES

The aim of the course is to introduce the basics of Computer Forensics including theoretical, technical, methodological and legal aspects. Moreover, the course presents the techniques of forensics investigation whenever digital crime evidences can be considered both specifically in the case of cybercrime and when in general digital data can be used as evidence for other types of crimes.

PROGRAM

Introduction to computer forensics. Methodological and critical aspects of computer forensics. Legal and practical aspects for producing evidence. Legal and technological aspects concerning the reliability of digital data, handling of digital evidence: more specifically the disk forensics and file systems handling for the proper information recovery. Hardware and Software tools used in the field of digital forensics, including the network forensics, mobile forensics and embedded forensics. Methodologies for handling encrypted data.

TEACHING MODALITIES

Lectures and exercises.

TEACHING MATERIALS

EXAM

The exam is given in form	Written and oral	X	Written only		Oral only	
In case of written exam, the tests are	Multiple choice	X	Open answer	X	Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Data Analytics

TITOLO INSEGNAMENTO IN INGLESE Data Analytics

Docente: Domenico Ciunzio

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
IINF-03/A	6	X			X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Lo studente acquisirà familiarità con le principali metodologie per l'estrazione di informazione da un insieme dati. Il percorso formativo comprende sia la teoria classica della stima/classificazione bayesiana e non sia la sua connessione alle più recenti tecniche di elaborazione e rappresentazione efficienti dei dati.

PROGRAMMA

Richiami di algebra lineare e calcolo delle probabilità.
 Teoria della stima. Generalità, parametri di qualità.
 Rappresentazione ed elaborazione dei dati: Statistica Sufficiente, Riduzione della dimensionalità, estrazione delle feature.
 Stima di parametri: Principio della massima verosimiglianza e Stima a massima verosimiglianza di modelli lineari. Metodo dei minimi quadrati.
 Stima bayesiana: Stima massima probabilità a posteriori (MAP). Stima a minimo errore quadratico medio (MMSE), Stima lineare MMSE.
 Classificazione: Generalità, parametri di qualità.
 Classificazione: non bayesiana: Classificazione a massima verosimiglianza, Criterio di Neyman-Pearson, Test di ipotesi composte.
 Classificazione bayesiana: regola MAP ed a massima verosimiglianza.

MODALITA' DIDATTICHE

Lezioni e piccoli progetti fatti da gruppi di lavoro con elaborati a calcolatore.

MATERIALE DIDATTICO

Appunti del corso:
 S.M. Kay: "Fundamentals of Statistical Signal Processing, Volume I e II: Estimation Theory ", Prentice Hall, 1993.
 S.M. Kay: "Fundamentals of Statistical Signal Processing, Volume II: Detection Theory ", Prentice Hall, 1998

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Data Analytics



Teacher: Domenico Ciunzio

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
IINF-03/A	6	X			X		

Propaedeutic courses: None

OBJECTIVES

The student will become familiar with the main methodologies for extracting information from a data set. The training course includes both the classical theory of estimation / classification (Bayesian and non-Bayesian), as well as its connection to the most recent efficient data processing and representation techniques.

PROGRAM

Review on Linear algebra and Probability Theory
 Estimation theory. Generality, quality indicators.
 Data representation and processing: Sufficient Statistics, Dimensionality reduction, feature extraction.
 Non-Bayesian estimation: Maximum likelihood principle and Maximum likelihood estimation of linear models. Least squares method.
 Bayesian estimation: Maximum posterior probability estimation (MAP). Minimum Quadratic Mean Error Estimation (MMSE), MMSE Linear Estimation.
 Classification: Generality, quality indicators.
 Non Bayesian Classification: Maximum likelihood classification, Neyman-Pearson criterion, Test of compound hypotheses.
 Bayesian Classification: MAP rule and maximum likelihood.

TEACHING MODALITIES

Lectures and small projects made by working groups.

TEACHING MATERIALS

Lecture notes:
 S.M. Kay: "Fundamentals of Statistical Signal Processing, Volume I e II: Estimation Theory ", Prentice Hall, 1993.
 S.M. Kay: "Fundamentals of Statistical Signal Processing, Volume II: Detection Theory ", Prentice Hall, 1998

EXAM

The exam is given in form	Written and oral	Written only	Oral only	X
In case of written exams, the tests are	Multiple choice	Open answer	Numeric	
Other				

SCHEDA DELL' INSEGNAMENTO DI Didattica dell'Informatica

TITOLO INSEGNAMENTO IN INGLESE Computer science education

Docente: da definire.

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

L'insegnamento introduce gli studenti alle varie problematiche relative alla formazione in ambito informatico e alle metodologie e tecniche per progettare percorsi disciplinari per l'insegnamento dell'informatica nella scuola secondaria di ii grado.

Gli studenti avranno:

- comprensione degli aspetti pedagogici fondamentali e delle teorie di apprendimento, con particolare attenzione all'ambito informatico
- comprensione delle linee guida e indicazioni nazionali dei curricula in informatica
- comprensione dei principali approcci e principi esistenti per la formazione
- consapevolezza delle questioni relative alla scelta del primo linguaggio di programmazione e dei corsi di introduzione all'informatica
- consapevolezza delle problematiche e degli approcci relativi alla verifica dell'apprendimento
- consapevolezza delle questioni relative alla diversità di genere

Capacità di applicare conoscenza e comprensione: essere in grado di sviluppare un percorso e le relative risorse per far acquisire conoscenze e abilità in ambito informatico.

PROGRAMMA

Da definire.

MODALITA' DIDATTICHE

Da definire.

MATERIALE DIDATTICO

Da definire.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	x	Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Computer science education

Teacher: to be defined.

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

The course introduces students to various issues related to computer science education and to the methodologies and techniques for designing curricula for teaching computer science in upper secondary school.

Students will acquire:

- an understanding of fundamental pedagogical aspects and learning theories, with particular focus on computer science
- familiarity with national guidelines and curricular frameworks for computer science education
- knowledge of the main existing approaches and principles for teaching computer science
- awareness of key issues related to the choice of an initial programming language and introductory computer science courses
- awareness of challenges and strategies related to assessment of learning
- sensitivity to issues concerning gender diversity

Ability to apply knowledge and understanding: students will be able to design a curriculum and related teaching resources aimed at developing knowledge and skills in the field of computer science.

PROGRAM

To be defined.

TEACHING MODALITIES

To be defined.

TEACHING MATERIALS

To be defined.

EXAM

The exam is given in form	Written and oral	x	Written only		Oral only	X
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Diritto dell'Informatica

TITOLO INSEGNAMENTO IN INGLESE Legal Informatics

Docente: da definire.

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
GIUR-17/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Obiettivo del corso è quello di fornire allo studente consapevolezza dell'esistenza di problemi giuridico/normativi legati alle nuove tecnologie, nonché strumenti conoscitivi per comprendere meglio il diritto d'autore, gestire progetti e attività professionali con un maggiore grado di autonomia, sviluppare e utilizzare tecnologie informatiche in modo conforme alla legge, ed avere consapevolezza degli aspetti giuridici in merito alla privacy in sistemi informatici.

PROGRAMMA

Introduzione al corso: l'obiettivo del corso è quello di indicare le principali modalità che danno rilevanza all'informatica nel settore giuridico.

Ci si soffermerà pertanto sui seguenti istituti:

Copyright/diritto d'autore. La disciplina del diritto d'autore. Software, banche di dati, opere multimediali, siti web. Software proprietario e open source. La titolarità del software creato e contratto di libera professione. I diritti musicali e cinematografici. Peer to peer. Digital Rights Management. Tutela dei domain names.

Proprietà industriale. Il codice della proprietà industriale. Marchi, nomi a dominio e brevetti. Tutela brevettuale del software. Le rivendicazioni. Le banche dati dei brevetti nazionali e internazionali.

Contratti software. Il contratto e gli elementi essenziali del contratto. Nullità e annullabilità. Contratti tipici e atipici. Principali tipologie di contratti utili per l'ingegnere. Cessione dei diritti di utilizzazione economica del software. Licenza d'uso di software, licenze open source. Creative commons. Contratti di sviluppo software.

Privacy e sicurezza. Privacy e protezione dei dati personali nella società dell'informazione. Principi generali. Informativa e consenso. Diritti dell'interessato. Dati sensibili. Misure di sicurezza minime e idonee. Privacy e big data. Profilazione degli utenti in rete e comunicazioni commerciali - Il Regolamento Generale europeo sulla protezione dei dati (GDPR 679/2016).

Firme elettroniche/digitali e documenti informatici. Le firme elettroniche: tipologie e disciplina giuridica. Il documento informatico: definizione, validità e rilevanza giuridica. La validità legale di e-mail, pagine web, registri informatici. La certificazione della firma digitale. Posta elettronica certificata. Conservazione sostitutiva dei documenti informatici - Codice dell'amministrazione digitale (D.Lgs. 82/2005).

Commercio elettronico. Obblighi e responsabilità dei prestatori di servizi on-line. Responsabilità degli intermediari. Comunicazioni commerciali. I contratti conclusi con strumenti informatici. La tutela dei consumatori in rete - Codice del consumo. Diritto di recesso e clausole vessatorie.

Reati informatici. Danneggiamento o interruzione di un sistema informatico. Interferenza illecita in comunicazioni informatiche o telematiche. Frode informatica.

MODALITA' DIDATTICHE

La didattica sarà articolata mediante lezioni frontali, esercitazioni seminariali anche con strumentazioni multimediali.

MATERIALE DIDATTICO

M. Farina, Elementi di diritto dell'informatica, Cedam, Milano, ultima edizione.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Discussione di elaborato progettuale.					

TEACHING SUMMARY OF THE COURSE Legal Informatics

Teacher: to be defined.

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
GIUR-17/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

The aim of the course is to provide students with an awareness of the legal and regulatory issues related to new technologies, as well as the knowledge needed to better understand copyright law, manage projects and professional activities with greater autonomy, develop and use information technologies in compliance with the law, and understand the legal aspects of privacy in computer systems.

PROGRAM

The aim of the course is to present the main ways in which computer science is relevant to the legal sector.

The course will therefore focus on the following topics:

Copyright. The legal framework of copyright law. Software, databases, multimedia works, websites. Proprietary and open-source software. Ownership of software created and freelance contracts. Musical and cinematographic rights. Peer-to-peer file sharing. Digital Rights Management. Protection of domain names.

Industrial Property. The Code of Industrial Property. Trademarks, domain names, and patents. Patent protection for software. Patent claims. National and international patent databases.

Software Contracts. Contracts and their essential elements. Nullity and annulment. Typical and atypical contracts. Main types of contracts relevant for engineers. Transfer of economic exploitation rights of software. Software license agreements, open-source licenses. Creative Commons. Software development contracts.

Privacy and Security. Privacy and the protection of personal data in the information society. General principles. Information notice and consent. Data subjects' rights. Sensitive data. Minimum and adequate security measures. Privacy and big data. User profiling and commercial communications – The European General Data Protection Regulation (GDPR 679/2016).

Electronic/Digital Signatures and Digital Documents. Electronic signatures: types and legal framework. Digital documents: definition, validity, and legal relevance. Legal validity of e-mails, web pages, and digital logs. Digital signature certification. Certified electronic mail. Electronic document retention – The Digital Administration Code (Legislative Decree 82/2005).

E-commerce. Obligations and liabilities of online service providers. Liability of intermediaries. Commercial communications. Contracts concluded via electronic means. Consumer protection online – Consumer Code. Right of withdrawal and unfair contract terms.

Computer Crimes. Damage to or disruption of computer systems. Unlawful interference with IT or telematic communications. Computer fraud.

TEACHING MODALITIES

Teaching will be delivered through lectures and seminar-based exercises, including the use of multimedia tools.

TEACHING MATERIALS

M. Farina, Elementi di diritto dell'informatica, Cedam, Milano, ultima edizione.

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Project work discussion.					

SCHEDA DELL' INSEGNAMENTO DI Encoding and Encryption

Docente: Alessandro De Luca

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso punta a introdurre i vari aspetti e scopi della codifica dei dati, quali la riduzione dei costi (compressione dei dati), affidabilità (correzione degli errori), e sicurezza (crittografia), il tutto nell'ambito della teoria dell'informazione di Shannon. Gli studenti padroneggeranno le tecniche basilari della teoria dei codici di sorgente e di canale, nonché gli algoritmi chiave per la crittografia classica e moderna (a chiave pubblica).

PROGRAMMA

Nozioni preliminari: Il modello di comunicazione di Shannon; Scopi della codifica; Parole e linguaggi formali; Codici (di sorgente) come linguaggi univocamente decodificabili; Codici prefissi e suffissi.

Codifica di sorgente: Caratterizzazioni di codici; Disuguaglianza di Kraft–McMillan; Codici massimali e completi; Codici prefissi massimali e rappresentazione con alberi; Sorgenti discrete a memoria 0 e loro entropia; Minimizzazione dei costi (teorema di Shannon sulla codifica di sorgente); Codici ottimali; Codici di Huffman; Ritardo di decifrazione e sincronizzazione.

Entropia di più variabili: Variabili aleatorie ed entropia; Entropia congiunta e condizionata; Mutua informazione; Disuguaglianze sull'elaborazione dei dati (DPI) e di Fano; Cenni a catene di Markov e sorgenti con memoria.

Codifica di canale: Canali discreti e loro capacità; Probabilità d'errore, tasso di codifica e sua raggiungibilità (teorema di Shannon sulla codifica di canale); Separazione sorgente–canale; Codici lineari e di Hamming; Cenni a codici ciclici e BCH.

Codifica crittografica: Introduzione e aritmetica di base; Crittosistemi a chiave simmetrica, esempi e crittoanalisi; Cifratura in termini di canale di comunicazione; Cifrario perfetto; Crittografia a chiave pubblica e RSA; Funzioni hash e autenticazione; Scambio di chiavi Diffie–Hellman; Cenni ad altri problemi «difficili» usati in crittografia (logaritmo discreto, curve ellittiche).

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

A. de Luca, F. D'Alessandro, *Teoria degli automi finiti*. Springer, 2013.

J. Berstel, D. Perrin, C. Reutenauer, *Codes and Automata*. Cambridge University Press, 2010.

T.M. Cover, J.A. Thomas, *Elements of Information Theory* (2nd ed.). Wiley, 2012.

N.L. Biggs, *Codes: An Introduction to Information Communication and Cryptography*. Springer, 2008.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Progetto					

TEACHING SUMMARY OF THE COURSE Encoding and Encryption

Teacher: Alessandro De Luca

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

The course aims to introduce all aspects and purposes of data encoding, such as economy (data compression), reliability (correction of errors), and security (cryptography), all within the framework given by Shannon's information theory. Students will master the basic techniques of (discrete) source and channel coding theory, as well as the key algorithms for classical and modern (public-key) cryptography.

PROGRAM

Preliminaries: Shannon's model for communication; Purposes of data encoding; Words and formal languages; (Source) codes as uniquely decodable languages; Prefix and suffix codes.

Source coding: Characterizations of codes; Kraft–McMillan generalized inequality; Maximal and complete codes; Maximal prefix codes and representation by trees; Discrete memoryless sources and their entropy; Minimizing costs (Shannon's source coding theorem); Optimal codes; Huffman codes; Deciphering and synchronization delay.

Entropy of several variables: Random variables and entropy; Joint and conditional entropy; Mutual information; Data processing inequality and Fano's inequality; Markov chains and sources with memory (sketch).

Channel coding: Discrete memoryless channels and their capacity; Error probability, encoding rate and its achievability (Shannon's channel coding theorem); Source–channel separation; Linear and Hamming codes; Cyclic and BCH codes (sketch).

Cryptographic coding: Introduction and basic arithmetics; Symmetric key cryptosystems: examples and cryptanalysis; Encryption in terms of communication channel; Perfect secrecy; Public key cryptography and RSA; Hashing and authentication; Diffie–Hellman key exchange; More "hard" problems (discrete logarithm, elliptic curves) and their use in cryptography (sketch).

TEACHING MODALITIES

Lectures.

TEACHING MATERIAL

A. de Luca, F. D'Alessandro, *Teoria degli automi finiti*. Springer, 2013.

J. Berstel, D. Perrin, C. Reutenauer, *Codes and Automata*. Cambridge University Press, 2010.

T.M. Cover, J.A. Thomas, *Elements of Information Theory* (2nd ed.). Wiley, 2012.

N.L. Biggs, *Codes: An Introduction to Information Communication and Cryptography*. Springer, 2008.

EXAM

The exam is given in form	Written and oral		Written only		Oral only	X
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	
Other	Project.					

SCHEDA DELL' INSEGNAMENTO DI Formal Methods for Strategic Reasoning

TITOLO INSEGNAMENTO IN INGLESE Formal Methods for Strategic Reasoning

Docente: tace per il solo a.a. 2025/2026

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Inglese
INFO-01/A	6		X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il ragionamento di strategia è un tema centrale in diversi settori dell'Informatica. Nell'Intelligenza Artificiale svolge il compito di progettare piani individuali o collettivi di conseguimento di obiettivi specifici da parte di agenti autonomi (i robot ad esempio). Nella verifica dei sistemi è usata per la sintesi automatica di programmi distribuiti che soddisfano per costruzioni proprietà desiderate o per verificare l'esistenza di equilibri di Nash. IL corso ha la finalità di introdurre lo studente ai concetti basilari di questo filone di ricerca attualmente attivo. Si inizierà con l'introduzione ai tre temi centrali - giochi, logica ed automi – evidenziando le loro interconnessioni. A partire da questi temi fondamentali verranno definite logiche specifiche adeguate ad esprimere ragionamento di strategia nei sistemi multi-agenti.

Risultati di apprendimento

- Acquisire un solido background nei giochi, logiche e automi.
- Familiarizzare con le logiche moderne per il ragionamento strategico.

PROGRAMMA

- Logica
 - Propositional logic
 - Linear-time Temporal Logic
 - Branching-time Temporal Logic
 - Quantified Branching-time Temporal Logic
- Giochi
 - Two-player games of finite duration
 - Two-player games of infinite duration
- Automi
 - Automi su parole finite
 - Automi su parole infinite
 - Automi su alberi

- Logiche per Strategic Reasoning
 - Multi-agent games
 - Alternating-time Temporal Logic
 - Strategy Logic
 - Extensions (imperfect information, probabilistic)
 - Tools for multi-agent strategic reasoning

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono (*)	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Presentazioni di articoli di ricerca					

TEACHING SUMMARY OF THE COURSE Formal Methods for Strategic Reasoning

Teacher: not active only for the academic year 2025/2026

SSD	CFU	Year			Semester		Language
		I	II		I	II	English
INFO-01/A	6		X			X	

Propaedeutic courses: None

OBJECTIVES

Strategic Reasoning is central in various fields of computer science. In Artificial Intelligence it aims at designing methods so that autonomous agents, such as robots, can make individual or collective plans to achieve their goals. In systems verification it is used to automatically synthesize distributed programs that are correct by construction, or to check the existence of Nash equilibria. This course constitutes an introduction to the main concepts involved in this active field of research. We will start with an introduction to the central trilogy of formal methods, i.e., games, logic and automata, as well as their deep connections. We will then build on this to present logics tailored to reason about strategies in multi-agent systems.

Learning Outcomes

- Acquire a solid background in games, logic and automata
- Familiarize with modern logics for strategic reasoning

PROGRAM

- Logic
 - Propositional logic
 - Linear-time Temporal Logic
 - Branching-time Temporal Logic
 - Quantified Branching-time Temporal Logic
- Games
 - Two-player games of finite duration
 - Two-player games of infinite duration
- Automata
 - Automata on finite words
 - Automata on infinite words
 - Automata on infinite trees
- Logics for Strategic Reasoning
 - Multi-agent games
 - Alternating-time Temporal Logic
 - Strategy Logic
 - Extensions (imperfect information, probabilistic)
 - Tools for multi-agent strategic reasoning

TEACHING MODALITIES

Lectures

TEACHING MATERIAL

--

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Discussion of research articles.					

SCHEDA DELL' INSEGNAMENTO DI Game Design and Development

TITOLO INSEGNAMENTO IN INGLESE Game Design and Development

Docente: Marco Faella

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	
INFO-01/A	6	X			X		Italiano, Inglese

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Al termine del corso, ci si aspetta che lo studente abbia acquisito le seguenti conoscenze ed abilità:

- conoscere le diverse tipologie di videogiochi
- conoscere gli elementi fondamentali che compongono l'esperienza di gioco
- saper strutturare un'applicazione grafica interattiva ad alta performance
- saper utilizzare una libreria di simulazione della meccanica dei corpi rigidi
- saper interagire programmaticamente con i sensori tipici dei device mobili
- saper ideare e realizzare semplici giochi per device mobili

PROGRAMMA

Storia e tipologie di videogiochi. Elementi di Game Design.

Richiami di Java. Android come ambiente di programmazione. Struttura di un game engine.

Tecniche di programmazione per la grafica e l'audio. Tecniche di programmazione per la simulazione della meccanica dei corpi rigidi. Uso programmatico di touch-screen e sensori di localizzazione.

Tecniche di programmazione ad alta performance.

Elementi di Intelligenza Artificiale per applicazioni ludiche

Sviluppo guidato di un progetto di videogioco.

MODALITA' DIDATTICHE

Lezioni frontali ed esercitazioni.

MATERIALE DIDATTICO

Robert Green e Mario Zechner, Beginning Android Games, Apress

Jesse Schell, The Art of Game Design, CRC Press

Ian Millington, Artificial Intelligence for Games, Morgan Kaufmann

Trasparenze disponibili sul sito del docente

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☐
In caso di prova scritta i quesiti sono (*)	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Sviluppo di un progetto software.					

TEACHING SUMMARY OF THE COURSE Game Design and Development



Teacher: Marco Faella

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian, English
INFO-01/A	6	X			X		

Propaedeutic courses: None

OBJECTIVES

The course aims at providing students with the following abilities:

- distinguishing various types of computer games
- describing the main aspects of the game experience, and their relationships with design objectives and techniques
- designing real-time interactive graphical applications
- using a 2D rigid-body physics simulation library
- programming with sensors on mobile devices
- deploying the appropriate AI component to achieve a design objective
- building a game prototype from scratch

PROGRAM

History and types of videogames. Elements of Game Design: design objectives and techniques.

Mobile operating systems and programming environments. Introduction to game engines.

2D graphics programming. Sound and music programming. Using a 2D rigid-body physics simulation library. Interfacing with a touch-screen and motion/localization sensors.

High-performance programming techniques. Architectures and patterns for games.

AI algorithms and techniques for games.

Guided development of a game prototype.

TEACHING MODALITIES

Lectures, seminars, and in-class exercises.

TEACHING MATERIALS

Robert Green e Mario Zechner, Beginning Android Games, Apress

Jesse Schell, The Art of Game Design, CRC Press

Ian Millington, Artificial Intelligence for Games, Morgan Kaufmann

Slides provided by the lecturer

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Development of a game prototype.					

SCHEDA DELL' INSEGNAMENTO DI Game Engines and Interactive Experience

TITOLO INSEGNAMENTO IN INGLESE Game Engines and Interactive Experience

Docente: Antonio Origlia

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano, Inglese
INFO-01/A	6		X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Al termine del corso, ci si aspetta che lo studente abbia acquisito le seguenti conoscenze ed abilità:

- Conoscere la struttura tipica e i servizi offerti dai moderni game engine
- Saper estendere un game engine con funzionalità custom o di terze parti
- Conoscere gli elementi di psicologia computazionale di base per lo stabilimento ed il mantenimento dell'engagement
- Comprendere il funzionamento dei giochi come mezzi di comunicazione e di persuasione
- Saper progettare sistemi di controllo del gioco coerenti con l'esperienza di gioco intesa
- Saper impiegare tecniche 3D avanzate in connessione allo stato interno del gioco
- Saper ottimizzare i contenuti multimediali per ridurre il carico computazionale

PROGRAMMA

- Struttura interna di un game engine
- Richiami di C++ e sviluppo di interfacce con motori di gioco industry-grade (ad es., Unreal Engine)
- Rappresentazione formale degli stati emotivi ed influenza nei meccanismi decisionali
- L'impatto dello storytelling nella progettazione dei sistemi di controllo e delle interfacce
- Tecniche di modellazione 3D ottimizzata per il rendering in tempo reale
- Sviluppo guidato di un progetto di videogioco desktop

MODALITA' DIDATTICHE

Lezioni frontali ed esercitazioni

MATERIALE DIDATTICO

Game engine architecture (Jason Gregory)
Digital storytelling (Carolyn Handler Miller)

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro	Sviluppo di un prototipo di gioco usando un game engine					

TEACHING SUMMARY OF THE COURSE Game Engines and Interactive Experience

Teacher: Antonio Origlia

SSD	CFU	Year			Semester		Language
		I	II		I	II	
INFO-01/A	6		X			X	Italian, English

Propaedeutic courses: None

OBJECTIVES

By the end of the course, students are expected to possess the following set of skills:

- Describing the typical structure and the main services offered by modern game engines
- Extending a game engine with custom or third-party features
- Employing the basic notions of computational psychology to establish and maintain engagement
- Analyzing and designing games as means of communication or persuasion
- Designing game control systems coherently with the intended experience
- Using advanced 3D techniques to represent the internal state of the game
- Optimizing media contents to reduce the computational load

PROGRAM

Internal structure of a game engine. C++ basics. and interfaces development with industry grade game engines (e.g. Unreal Engine).

Formal representation of emotional states and their influence in decision making. The impact of storytelling in designing control and interface systems.

3D modelling techniques optimized for real-time rendering.

Guided development of a desktop game.

TEACHING MODALITIES

Lectures, seminars, and in-class exercises.

TEACHING MATERIALS

Game engine architecture (Jason Gregory)

Digital storytelling (Carolyn Handler Miller)

EXAM

The exam is given in form	Written and oral	Written only	Oral only	X
In case of written exams, the tests are	Multiple choice	Open answer	Numeric	
Other	Development of a game prototype using a game engine			

SCHEDA DELL'INSEGNAMENTO DI Human-Robot Interaction

Docente: Silvia Rossi

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

L'utilizzo di Robot si sta diffondendo dalle industrie ai domini civili quali case, uffici, scuole e ospedali. Come deve essere progettato un sistema robotico destinato all'interazione con l'utente in tali domini?

L'obiettivo di questo corso è quello di porre enfasi sulla progettazione dell'Interazione Uomo-Robot (Human-Robot Interaction – HRI), applicata sia a robot mobili che a umanoidi. In particolare, i metodi informatici e sperimentali dell'HRI saranno alla base dei concetti che gli studenti andranno ad apprendere. Parte consistente di questo corso prevedrà l'implementazione di tali concetti su piattaforme robotiche reali, per cui gli studenti dovranno progettare, implementare e testare un sistema robotico in grado di interagire con l'essere umano.

Obiettivi del corso

- Prendere confidenza con le principali tecniche di percezione artificiale usate in HRI
- Implementare un sistema di interazione uomo-robot
- Pianificare e svolgere uno studio orientato all'utente
- Analizzare i risultati di uno studio

PROGRAMMA

- Introduzione: Interazione Uomo-Robot (HRI) (1 lezione)
- Azioni Reattive (6 lezioni)
 - Comportamenti ed etologia animale
 - Riflessi
 - Comportamenti sociali
 - Coordinazione e controllo nei behavior
 - Meccanismo innato di rilascio e Schema Theory
 - Architettura a Sussunzione
 - Architettura a campi potenziali
 - Progettare un Sistema reattivo con behavior
- Percezione in HRI (6 lezioni)
 - Introduzione a sensori e percezione
 - Percezione in behavior

- Ciclo percezione-azione
- Approccio ecologico di Gibson
- Sistemi percettivi di Neisser
- Riconoscimento e monitoraggio dell'utente
- Riconoscimento del parlato e del dialogo
- Riconoscimento di intenzioni
- Tecconologie affettive
- Azioni Intenzionali (3 lezioni)
 - Teoria delle Mente (ToM) e cambio di prospettiva
 - Prossemica e navigazione
 - Espressione delle intenzioni
 - Leggibilità del comportamento del robot
 - Gioco di squadra e coordinamento delle azioni
- Behavior non verbali in HRI (2 lezioni)
 - Emotional Intelligence
 - Gesti e linguaggio del corpo
 - Gaze e contatto visivo
 - Interazione multimodale
- Applicazioni di Robotica Sociale (2 lezioni)
 - Accettazione/approvazione e fiducia
 - Sviluppo di relazioni a lungo termine
- Esperimenti di HRI (4 lezioni)
 - Progettazione dell'interazione
 - Metodologia di ricerca in HRI
 - Teleoperazione e WoZ
 - Metriche e misure
 - Statistica inferenziale
 - T-test
 - Modello Lineare Generale
 - Analisi della varianza

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono (*)	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Progetto di HRI					

TEACHING SUMMARY OF THE COURSE Human-Robot Interaction

Teacher: Silvia Rossi

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

Robots are moving from factory floors, battlefields, and space into human environments such as homes, offices, schools, and hospitals. How can we design robotic systems for human interaction? This course is created to have a special emphasis on HRI design as it applies to mobile and humanoid robots. Students learn about the core computational and experimental methods in human-robot interaction (HRI). A big part of this course involved the implementation of the HRI concepts on an actual robot platform. Students also learn how to design and analyze experiments to evaluate HRI systems.

Learning Outcomes

- Familiarize with the main perception techniques for HRI
- Implement a human-robot interaction system
- Plan and execute a human-subject study
- Analyze the results of a study using inferential statistics

PROGRAM

- Introduction: Human-Robot Interaction (HRI) (1 lezione)
- Reactive Actions (6 lezioni)
 - Animal Behaviors and Ethology
 - Reflexes and Taxes
 - Social Behaviors
 - Coordination and control in behaviors
 - Innate releasing mechanism and Schema Theory
 - Subsumption Architecture
 - Potential Field Architecture
 - Designing a reactive system with behaviors
- Perception in HRI (6 lezioni)
 - Introduction on sensors and perception
 - Perception in behaviors
 - Perception-action cycle
 - Gibson ecological approach

- Neisser two perceptive systems
- Human recognition and tracking
- Speech recognition and dialogue
- Intention Recognition
- Affective technologies
- Intentional Action (3 lezioni)
 - ToM and Perspective Taking
 - Proxemics and Navigation
 - Intention Expression
 - Legible Robot Motion
 - Teamwork and Action Coordination
- Nonverbal Behavior in HRI (2 lezioni)
 - Emotional Intelligence
 - Gestures and Body Language
 - Gaze and Eye Contact
 - Multimodal Interaction
- Social Robotics Applications (2 lezioni)
 - Acceptance and Trust
 - Developing long-term relationships
- HRI Experiments (4 lezioni)
 - Interaction Design
 - Research Methods in HRI
 - Teleoperation and WoZ
 - Metrics and Measurement
 - Inferential Statistics
 - T-test
 - General Linear Model (GLM)
 - Analysis of Variance (ANOVA)

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Project development.					

SCHEDA DELL' INSEGNAMENTO DI Intelligent Robotics

TITOLO INSEGNAMENTO IN INGLESE Intelligent Robotics

Docente: Alberto Finzi

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso introduce gli studenti alle basi teoriche e alle metodologie necessarie per la progettazione di sistemi robotici autonomi/cognitivi in grado di operare in ambienti non strutturati, dinamici ed interattivi. La prima parte del corso è focalizzata su concetti e metodi per la robotica mobile (localizzazione, mapping, navigazione, esplorazione, ecc.). La seconda parte è incentrata su architetture e metodi per la progettazione di sistemi robotici autonomi dotati di capacità cognitive (deliberazione, pianificazione, interazione, collaborazione, apprendimento, ecc.). Al termine del corso gli studenti avranno acquisito: conoscenza e comprensione delle problematiche e degli approcci proposti in letteratura; competenza nelle architetture, nei modelli, nei metodi e nelle tecniche necessari per la progettazione di sistemi robotici autonomi/cognitivi; competenza negli strumenti per lo sviluppo di robot autonomi.

PROGRAMMA

Robotica Mobile: (3 CFU) Paradigmi e architetture; introduzione alla robotica mobile; filtri bayesiani (filtri di Kalman e filtri particellari); metodi per la localizzazione e il mapping (EKF-SLAM, FastSLAM); navigazione (evitamento di ostacoli, metodi dynamic windows); pianificazione di percorso e di moto (metodi Voronoi, PRM, RRT); metodi di pianificazione probabilistica (MDPs, POMDPs); apprendimento per rinforzo in robotica.

Robotica Autonoma e Cognitiva: (3 CFU) architetture ibride; middleware e integrazione (introduzione a ROS); pianificazione classica; pianificazione gerarchica; pianificazione e schedulazione temporale; sistemi esecutivi; architetture cognitive e modelli cognitivi; metodi per la robotica collaborativa (pianificazione ed esecuzione collaborativa, sistemi human-in-the-loop e autonomia regolabile, apprendimento per dimostrazione).

MODALITA' DIDATTICHE

Lezioni frontali e attività/esercitazioni di laboratorio.

MATERIALE DIDATTICO

Probabilistic Robotics, Sebastian Thrun, Wolfram Burgard and Dieter - MIT Press
Introduction to AI Robotics, R. Murphy - MIT Press
Automated Planning and Acting, Ghallab, Nau, Traverso, Cambridge University Press
Documenti, lucidi e note forniti sul sito web del corso

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	X	Solo orale	☐
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	X	Esercizi numerici	☐
Altro	Progetto					

TEACHING SUMMARY OF THE COURSE Intelligent Robotics

Teacher: Alberto Finzi

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X		-	X	

Insegnamenti propedeutici previsti: nessuno

OBJECTIVES

The course introduces students to the theoretical foundations and the methodologies needed for the design of autonomous/cognitive robotic systems able to operate in unstructured and dynamic environments. The first part of the course focuses on concepts and methods for mobile robotics (localization, mapping, navigation, exploration, etc.). The second part is focused on architectures and methods for the design of autonomous robotics systems endowed with cognitive capabilities (deliberation, planning, interaction, collaboration, learning, etc.). Upon completion of the course students will be provided with knowledge and understanding of the problems and the approaches proposed in the literature; competence in the architectures, models, methods and techniques needed for the design of autonomous/cognitive robotic systems; competence in tools for autonomous robots design and development.

PROGRAM

Mobile Robotics: (3 CFU) Paradigms and architectures; introduction to mobile robotics; bayesian filters (kalman filters and particle filters); methods for localization and mapping (EKF-SLAM, FastSLAM); navigation (obstacle avoidance, dynamic windows methods); path and motion planning (Voronoi methods, PRM, RRT); probabilistic planning methods (MDPs, POMDPs); reinforcement learning in robotics.

Autonomous and Cognitive Robotics: (3 CFU) hybrid architectures; middleware and integration (introduction to ROS); classical planning; hierarchical planning; temporal planning and scheduling; executive systems; cognitive architectures and cognitive models; methods for collaborative robotics (collaborative planning and execution, human-in-the-loop systems and sliding autonomy, learning by demonstration).

TEACHING MODALITIES

Frontal lectures and lab activities/exercises.

TEACHING MATERIALS

Probabilistic Robotics, Sebastian Thrun, Wolfram Burgard and Dieter - MIT Press
 Introduction to AI Robotics, R. Murphy - MIT Press
 Automated Planning and Acting, Ghallab, Nau, Traverso, Cambridge University Press
 Additional documents and lecture notes provided on the course website

EXAM

The exam is given in form	Written and oral		Written only	X	Oral only	
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	
Other	Project					

SCHEDA DELL' INSEGNAMENTO DI Intelligent Web



TITOLO INSEGNAMENTO IN INGLESE Intelligent Web

Docente: Luigi Sauro

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Scopo del corso è quello di approfondire le principali tematiche fondazionali, metodologiche e applicative che riguardano l'uso dell'Intelligenza Artificiale nel rappresentare, accumulare, integrare e rintracciare informazioni sul Web. In particolare, lo studente affronterà lo studio dei principali linguaggi e tecnologie dei Linked Data e del Semantic Web. Il corso consentirà di acquisire le competenze opportune per sviluppare ontologie e approfondirà lo studio delle tecniche di ragionamento automatico necessarie per analizzare i costi computazionali di alcuni ontology design patterns. Il corso si occuperà anche del problema di riconoscere le preferenze di un utente attraverso tecniche di apprendimento attivo basate su inferenze Bayesiane.

PROGRAMMA

Linked Data: modelli No-SQL per i dati sul Web, RDF, RDF-Schema, SPARQL. Strumenti di sviluppo di repositories RDF, GraphDB. Casi di studio, CIDOC CRM e DBpedia. Semantic Web: Ontology Web Language (OWL2), sintassi e semantica. Ontology design patterns e strumenti di sviluppo e analisi di ontologie, Protégé. Fondamenti logici dei task di ragionamento nel Semantic Web, Description Logics e tableaux systems. Profili di OWL2 a bassa complessità (OWL-EL, OWL-DL, and OWL-RL). Bayesian Preference Elicitation, nozione di belief improvement e sue tecniche di misura. Problemi di carattere computazionale nei recommender systems basati su Preference Elicitation.

MODALITA' DIDATTICHE

Lezioni in aula. Sviluppo di progetti di gruppo.

MATERIALE DIDATTICO

An Introduction to Description Logics, Franz Baader, Technische Universität, Dresden, Ian Horrocks, University of Oxford, Carsten Lutz, Universität Bremen, Uli Sattler, University of Manchester. Cambridge University Press.

The Description Logic Handbook, Franz Baader, Technische Universität, Dresden, Diego Calvanese, Freie Universität Bozen, Bolzano, Deborah L. McGuinness, Rensselaer Polytechnic Institute, New York, Daniele Nardi, Università degli Studi di Roma 'La Sapienza', Italy, Peter F. Patel-Schneider, AT&T Bell Laboratories, New Jersey. Cambridge University Press.

Semantic Web for the Working Ontologist. Dean Allemang, James Hendler, Elsevier.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☐
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☒
Altro	Progetto					

TEACHING SUMMARY OF THE COURSE Intelligent Web



Teacher: Luigi Sauro

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X		X		

Propaedeutic courses: None

OBJECTIVES

The aim of the course is to explore the main foundational, methodological, and applicative issues underlying the use of Artificial Intelligence technologies to represent, gather, merge and retrieve information on the Web. Specifically, the student is introduced to the main languages and technologies concerning Linked Data and the Semantic Web. The course will be focused on providing the appropriate skills to develop principled ontologies and a detailed study of automated reasoning methodologies that are necessary to analyze the computational effects of some ontology design patterns. The course will also address the problem of understanding the preferences of a user by using active learning methodologies based on Bayesian inferences.

PROGRAM

Linked Data: No-SQL models of data on the Web, RDF, RDF-Schema, SPARQL. Tools to develop RDF repositories, GraphDB. Case studies, CIDOC CRM and DBpedia. Semantic Web: Ontology Web Language (OWL2), syntax and semantics. Ontology design patterns, and tools to develop and analyze ontologies, Protégé. Logical foundations of the Semantic Web reasoning tasks, Description Logics and tableaux systems. Low-complexity OWL2 profiles (OWL-EL, OWL-DL, and OWL-RL). Bayesian Preference Elicitation, the notion of belief improvement and its measures. Computational issues on PE-based recommender systems.

TEACHING MODALITIES

Lectures. Development of a group project.

TEACHING MATERIAL

An Introduction to Description Logics, Franz Baader, Technische Universität, Dresden, Ian Horrocks, University of Oxford, Carsten Lutz, Universität Bremen, Uli Sattler, University of Manchester. Cambridge University Press.

The Description Logic Handbook, Franz Baader, Technische Universität, Dresden, Diego Calvanese, Freie Universität Bozen, Bolzano, Deborah L. McGuinness, Rensselaer Polytechnic Institute, New York, Daniele Nardi, Università degli Studi di Roma 'La Sapienza', Italy, Peter F. Patel-Schneider, AT&T Bell Laboratories, New Jersey. Cambridge University Press.

Semantic Web for the Working Ontologist. Dean Allemang, James Hendler, Elsevier.

EXAM

The exam is in the form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	X
Other	Project					

SCHEDA DELL' INSEGNAMENTO DI Logics for Computer Science



TITOLO INSEGNAMENTO IN INGLESE Logic for Computer Science

Docente: Massimo Benerecetti

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Acquisire una conoscenza delle principali proprietà sintattiche e semantiche della logica classica proposizionale e della logica del primo ordine. Acquisire familiarità con i principali sistemi deduttivi della logica classica che sono di interesse per l'informatica. Acquisire la capacità di formalizzare enunciati dichiarativi, problemi e ragionamenti nel linguaggio della logica classica, nonché di verificare la correttezza di ragionamenti informali.

PROGRAMMA

Logica proposizionale classica. Nozioni sintattiche di base. Nozioni semantiche: soddisfacibilità, tautologie, conseguenza logica e insoddisfacibilità. Forme normali congiuntive e disgiuntive. Deduzione naturale. Calcolo dei sequenti. Tableaux analitici. Regola di risoluzione, procedura di Davis-Putnam, metodo refutazionale. Correttezza, completezza e compattezza della logica proposizionale.

Logica classica del primo ordine. Elementi di sintassi. Semantica tarskiana: soddisfacibilità, verità, validità, conseguenza logica. Tableaux analitici. Formalizzazione di ragionamenti informali. Clausole, universo di Herbrand, clausole ground e refutazioni di insiemi di clausole. Forma normale prenessa e skolemizzazione. Teorema di Skolem. Correttezza, completezza e compattezza della logica del primo ordine. Modelli non-standard dell'aritmetica. Cenni al teorema di indecidibilità della logica del primo ordine e ai teoremi di incompletezza di Goedel. Dimostrabilità, verità e insiemi ricorsivamente enumerabili.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Logics for Computer Science



Teacher: Massimo Benerecetti

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

Students will become acquainted with the main syntactic and semantic notions and properties of both classical propositional logic and classical first-order logic.

Students will additionally become familiar with the main deductive systems for both classical propositional logic and classical first-order logic and their roles in computational methodologies and systems.

Students will finally become capable to formalize natural language declarative statements in classical logic, and to verify the correctness of informal arguments.

PROGRAM

Classical propositional logic. Basic syntactic notions. Semantical notions: truth-value assignments, satisfiability, logical consequence, truth and logical validity. Conjunctive and disjunctive normal forms. Natural deduction. Sequent calculus. Analytic tableaux. Resolution, Davis-Putnam procedure, refutations of CNF formulas. Correctness, completeness and compactness of classical propositional logic.

Classical first-order logic. Basic syntactic notions. Tarskian semantics. Analytic tableaux. Formalization of natural language statements and the correctness of informal arguments. Clauses, Herbrand universes, ground clauses, refutations of sets of clauses. Prenex normal form. Skolemization. Skolem's theorem. Correctness, completeness and compactness of first-order logic. Non-standard models of arithmetic. Remarks on undecidability of first-order logic and Goedel's incompleteness theorems. Provability, truth and recursively enumerable sets.

TEACHING MODALITIES

Lezioni frontali. Esercitazioni.

TEACHING MATERIALS

EXAM

The exam is given in form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice		Open answer	X	Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Machine Learning (Neural Networks and Deep Learning)



TITOLO INSEGNAMENTO IN INGLESE Machine Learning. Neural Networks and Deep Learning

Docente: Roberto Prevete

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X			X		

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Obiettivo del corso è quello di introdurre gli studenti ai principali aspetti, sia teorici sia pratici, riguardanti la progettazione e l'addestramento di reti neurali artificiali sia feed-forward sia ricorrenti, focalizzando l'attenzione su alcuni task specifici come quello della classificazione delle immagini e del testo. Il corso, inoltre, fornisce una introduzione ai modelli di rete neurale profonda, come le reti convoluzionali, considerando alcune delle architetture che hanno avuto particolare successo. Il corso, per di più, vuole rendere gli studenti capaci di usare alcune delle principali librerie software disponibili per lo sviluppo e l'apprendimento delle reti neurali artificiali.

PROGRAMMA

Architetture di rete feed-forward. Funzioni di attivazione. Funzioni di errore. Metodo di back-propagation per il calcolo del gradiente. Regole di apprendimento basate sulla discesa del gradiente. Strategie di generalizzazione e di apprendimento. Regularizzazione. Rete neurale profonda: autoencoders a pila, unità rettificata, reti convoluzionali. Neurone biologico e modelli storicamente importanti: modelli di McCulloch-Pitts e Hodgkin-Huxley. Reti ricorrenti e ricorsive: reti neurali ricorrenti a tempo continuo, architetture Jordan ed Elman. Apprendimento per reti ricorrenti. Applicazioni di reti neurali a testi, immagini e serie storiche. Attuali aree di ricerca.

MODALITA' DIDATTICHE

Lezioni frontali e progetti di gruppo.

MATERIALE DIDATTICO

I libri di testo principali sono:

Christopher M. Bishop. Neural Networks for Pattern Recognition. Oxford University Press, 1996.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016.

Ulteriore materiale consiste in articoli scientifici recuperabili su librerie on-line, o forniti dal docente.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro	Sviluppo progetto					

TEACHING SUMMARY OF THE COURSE Machine Learning (Neural Networks and Deep Learning)



Teacher: Roberto Prevete

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X			X		

Propaedeutic courses: None

OBJECTIVES

The course aims to introduce the students to the main theoretical and applicative aspects regarding how to design and train feed forward and recurrent neural networks for tasks such as text classification and image recognition. This course, moreover, provides an introduction to deep neural network models, such as convolutional neural networks, and gives an overview of deep network architectures which have been particularly successful. The course also provides an introduction to the use of some of the software libraries available for building and training shallow and deep neural networks.

PROGRAM

Feed-forward network architectures. Activation functions. Loss functions. Back-propagation method. Learning rules based on gradient descend. Generalization and learning strategies. Regularization. Deep neural network: stacked autoencoders, rectified units, convolutional networks. Biological neurons and historically important models: McCulloch-Pitts and Hodgkin-Huxley models. Recurrent and recursive networks: continuous time recurrent neural networks, Jordan and Elman architectures. Applications of neural networks to text, images and time series. Current areas of research.

TEACHING MODALITIES

Lectures. Development of a group project.

TEACHING MATERIALS

The main books for the course are:

- Christopher M. Bishop. Neural Networks for Pattern Recognition. Oxford University Press, 1996.
- Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016.

Additional reading consists of scientific papers retrievable on public on-line libraries or made available by the teacher.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exam, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Project development					

SCHEDA DELL' INSEGNAMENTO DI Machine Learning (Statistical Learning)



TITOLO INSEGNAMENTO IN INGLESE Machine Learning. Statistical Learning

Docente: Anna Corazza

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Il corso mira a fornire agli studenti le competenze teoriche e computazionali necessarie per comprendere, progettare e attuare moderni approcci statistici di machine learning. Tali sistemi sono ampiamente adottati in diversi settori ed è importante essere in grado di formulare correttamente il problema e scegliere l'approccio più efficace per la situazione specifica.

PROGRAMMA

Definizioni di base dell'apprendimento automatico con e senza supervisione. Problema della valutazione degli algoritmi di apprendimento automatico: parametri di valutazione e protocolli sperimentali. Il modello a spazio vettoriale e la definizione di somiglianza. Feature selection. Classificazione: Naive Bayes, Rocchio, k Nearest Neighbour, support vector machines. Cluster: K-mean, approcci gerarchici. Etichettatura dei cluster. Gli esempi saranno preferibilmente presi dal campo dell'information retrieval.

MODALITA' DIDATTICHE

Lezioni frontali

MATERIALE DIDATTICO

Shai Shalev-Shwartz, Shai Ben-David, UNDERSTANDING MACHINE LEARNING: From Theory to Algorithms

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Machine Learning (Statistical Learning)



Teacher: Anna Corazza

SSD	CFU	Year			Semester		Language
		I	II		I	II	
INFO-01/A	6	X				X	Italian

Propaedeutic courses: None

OBJECTIVES

The course aims at providing the students with the necessary theoretical and computational skills to understand, design and implement modern statistical machine learning approaches. Such systems are widely adopted in different fields, and it is important to be able to give a correct formulation of the problem and to choose the most effective approach for the specific situation.

PROGRAM

Basic definitions of supervised and unsupervised machine learning. Assessment of a machine learning algorithm: evaluation parameters and experimental protocols. The vector space model and the similarity definition. Feature selection. Classification: Naive Bayes, Rocchio, k-Nearest neighbour, support vector machines. Clustering: K-means, hierarchical approaches. Cluster labeling. Examples will be preferably taken from the information retrieval field.

TEACHING MODALITIES

Lessons.

TEACHING MATERIAL

Shai Shalev-Shwartz, Shai Ben-David, UNDERSTANDING MACHINE LEARNING: From Theory to Algorithms

EXAM

The exam is given in form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice		Open answer	X	Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Mathematics for cryptography

TITOLO INSEGNAMENTO IN INGLESE Mathematics for cryptography

Docente: Maurizio Laporta

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
MATH-03/A	6		X		X		

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Scopo del corso è introdurre gli studenti ad argomenti di teoria dei numeri, sia antichi che molto moderni, i quali sono al centro dell'interesse nella crittografia contemporanea, soprattutto nei più noti crittosistemi a chiave pubblica quali RSA; si tiene un approccio algoritmico, ponendo l'accento su stime dell'efficienza delle tecniche derivanti dalla teoria.

PROGRAMMA

1. Teoria elementare dei numeri: notazioni e proprietà di base; divisibilità e algoritmo euclideo; congruenze; aritmetica modulare; funzioni aritmetiche di base; teorema cinese del resto; congruenze polinomiali modulo un numero primo; residui quadratici; simbolo di Legendre; simbolo di Jacobi; legge di reciprocità quadratica; campi finiti. 2. Teoria computazionale dei numeri: stime temporali per aritmetica elementare; nozioni di base su complessità computazionale e classificazione degli algoritmi; stima del numero di operazioni su bit necessarie ad eseguire al computer operazioni di teoria dei numeri, quali l'algoritmo euclideo, il metodo del ripetuto elevamento al quadrato e l'algoritmo di Jacobi; il problema del logaritmo discreto; la distribuzione dei numeri primi con applicazioni alla complessità computazionale. 3. Primalità: pseudoprimi; test di primalità (Solovay–Strassen e Miller–Rabin); stime temporali per test di primalità. 4. Fattorizzazione: nozioni di base sul problema della fattorizzazione; metodo di Eratostene; metodo di Fermat; metodo di Pollard; numeri smooth; metodo del crivello quadratico. 5. Aritmetica delle curve ellittiche: nozioni di base sulle curve ellittiche; test di primalità; metodo di Lenstra; problema del logaritmo discreto (DLP) sulle curve ellittiche. 6. Crittografia: chiavi simmetriche; crittografia a chiave pubblica; problema di Diffie–Hellman; protocollo RSA; crittosistemi basati su curve ellittiche.

MODALITA' DIDATTICHE

Lezioni ed esercitazioni

MATERIALE DIDATTICO

N.Koblitz, A Course in Number Theory and Cryptography, Springer Verlag
N.Koblitz, Algebraic Aspects of Cryptography, Springer Verlag
A. Languasco & A. Zaccagnini, Introduzione alla Crittografia, Hoepli
D.R. Stinson, Cryptography: Theory and Practice, Chapman and Hall Ed.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☐
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☒	Esercizi numerici	☒
Altro						

TEACHING SUMMARY OF THE COURSE Mathematics for cryptography

Teacher: Maurizio Laporta

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
MATH-03/A	6		X		X		

Propaedeutic courses: None

OBJECTIVES

Course aims: the purpose of the course is to introduce the student to number theoretic topics, both ancient and very modern, which are at the center of interest in contemporary cryptography, especially in the most known public key cryptosystems such as RSA; an algorithmic approach is taken, emphasizing estimates of the efficiency of the techniques that arise from the theory.

PROGRAM

Course Description: 1. Elementary Number Theory: notation and basic properties; divisibility and the Euclidean algorithm; congruences; modular arithmetic; basic arithmetic functions; the Chinese Remainder Theorem; polynomial congruences modulo a prime number; quadratic residues; the Legendre symbol; the Jacobi symbol; quadratic reciprocity law; finite fields 2. Computational Number Theory: times estimates for elementary arithmetic; basic notions on computational complexity and classification of the algorithms; estimating the number of bit operations needed to perform some number theoretic tasks by computer, such as the Euclidean algorithm, the repeated squaring method and the Jacobi algorithm; the discrete logarithm problem; the distribution of prime numbers with applications to the computational complexity. 3. Primality: pseudoprimes; primality tests (Solovay-Strassen and Miller-Rabin); times estimates for primality tests. 4. Factoring: basic facts on the factoring problem; the Eratosthenes method; the Fermat method; the Pollard method; smooth numbers; the quadratic sieve method. 5. Arithmetic of the Elliptic Curves: basic facts on the elliptic curves; primality test; the Lenstra method; DLP on the elliptic curves. 6. Cryptography: symmetric keys; public key cryptography; the Diffie-Hellman problem; the RSA protocol; elliptic curve cryptosystems.

TEACHING MODALITIES

Lectures.

TEACHING MATERIAL

N.Koblitz, A Course in Number Theory and Cryptography, Springer Verlag
 N.Koblitz, Algebraic Aspects of Cryptography, Springer Verlag
 A. Languasco & A. Zaccagnini, Introduzione alla Crittografia, Hoepli
 D.R. Stinson, Cryptography: Theory and Practice, Chapman and Hall Ed.

EXAM

The exam will be given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, tests are	Multiple choice	☐	Open answer	☒	Numeric	☒
Other						

SCHEDA DELL' INSEGNAMENTO DI Methods for Artificial Intelligence

TITOLO INSEGNAMENTO IN INGLESE Methods for Artificial Intelligence

Docente: Silvia Rossi

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II	III	I	II	Inglese
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

L'obiettivo di questo corso è di fornire agli studenti una conoscenza completa e approfondita dei principi e delle tecniche dell'intelligenza artificiale introducendo i problemi classici dell'IA, nonché i modelli e gli algoritmi utilizzati per affrontare questi problemi. Il programma del corso è diviso in tre parti. Nella prima saranno presentati algoritmi per la risoluzione di problemi di ricerca informata nello spazio degli stati, ricerca online ed in presenza di avversario, e problemi di soddisfacimento di vincoli. Nella seconda parte sarà analizzato il ragionamento e processo decisionale in caso di incertezza, discusso come rappresentare la conoscenza, inclusa la conoscenza incompleta e incerta del mondo reale; come ragionare logicamente con quella conoscenza usando le probabilità; come utilizzare questi modelli e metodi di ragionamento per decidere cosa fare. Nella terza parte saranno introdotte le problematiche relative a modelli distribuiti di decisione. In particolare, saranno presentati approcci di teoria dei giochi per la modellazione di processi di decisione nel caso di interazioni non cooperative e le possibili applicazioni di tali tecniche a problematiche concrete.

PROGRAMMA

Introduzione al Corso

- Cenni Storici

Problem Solving

- Tipologia di problemi: Problemi a Stato singolo, Problemi a stato multiplo, Problemi dipendenti da situazioni, Imprevedibili a priori, Problemi di tipo esplorazione.
- Strutture di rappresentazione e formalizzazione del problema per le prime tre tipologie.
- Strategie di ricerca non informata per le prime due tipologie di ricerca:
 - Ricerca in ampiezza e a costo uniforme;
 - Ricerca in profondità, e con iterazione della profondità;
 - Proprietà e confronto delle strategie di ricerca.
- Strategie di ricerca euristiche per le prime due tipologie di ricerca:
 - Greedy best-first, A* ed euristiche ammissibili;
 - Euristiche consistenti;
 - Iterative Deepening A*;

- Algoritmi di ricerca locale: Hill-climbing search, Local beam search, Simulated Annealing and Genetic.
- Giochi a più agenti: giochi a somma zero visti come problemi da risolvere;
 - Funzioni di utilità e strategia minimax;
 - Alpha beta pruning.
- Soluzione di problemi con vincoli: CSP
 - CSP come problema di ricerca;
 - Backtracking per CSP, ordinamento di variabili e valori;
 - Propagazione dei vincoli: Forward Checking e AC3;
 - Struttura del problema;
 - Ricerca locale.

Ragionamento e rappresentazione con conoscenza incerta

- Rappresentazione dell'incertezza:
 - Introduzione a decision theory, nozioni base di probabilità, eventi atomici, probabilità incondizionata, Joint probability distribution, probabilità condizionata;
 - Indipendenza e indipendenza condizionata, teorema di Bayes;
- Reti bayesiane:
 - Introduzione e tabelle delle probabilità condizionate;
 - Rappresentazione delle relazioni e delle probabilità condizionate;
 - Inferenze esatte: enumerazione, eliminazione di variabili.
- Ragionamento in condizioni di incertezza:
 - Azioni e utilità attesa, decision networks e rappresentazione di un problema di decisione, valore dell'informazione;
 - Problemi con decisioni in sequenza, utilità degli stati e equazione di Bellman, policy, soluzione con iterazione del valore, soluzione con iterazione della policy.

Sistemi multi-agente e game theory

- Introduzione ai sistemi multi-agente;
- Introduzione al game theory:
 - Giochi in forma normale e soluzioni
 - Strategie pure e miste; equilibri di Nash, Pareto ottimalità, social welfare;
 - Giochi a somma zero e rappresentazione estesa;
 - Calcolo di equilibri e dominanze;
 - Soluzioni di giochi a somma zero, minimax e alfabetà.
 - Giochi ripetuti.

- Computational social choice:
 - Meccanismi di votazione: funzioni di scelta sociale e social welfare;
 - Proprietà e teorema di Arrow;
 - Muller-Satterthwaite theorem;
 - Funzioni di ranking e PageRank;
 - Votazioni Strategiche e mechanism design (strategie dominanti e Bayes Nash);
- Allocating scarce resources:
 - Meccanismi d'asta, aste per item individuali (asta Inglese, asta Olandese, asta Giapponese, Vickrey);
 - Strategie dominanti ed equilibri nei meccanismi d'asta;
- Reaching agreements:
 - Negoziazione, domini;
 - Negoziazione in domini task oriented;
 - Approcci game teoretici, protocolli con offerte alternate;
 - Approcci euristici, protocollo di concessione monotona, strategia di Zeuthen;
 - Negoziazione multi-issue.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono (*)	A risposta multipla		A risposta libera	X	Esercizi numerici	X
Altro						

TEACHING SUMMARY OF THE COURSE Methods for Artificial Intelligence



Teacher: Silvia Rossi

SSD	CFU	Year			Semester		Language
		I	II		I	II	English
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

The objective of this course is to provide the students with a full and comprehensive knowledge of AI methods and techniques. We will introduce classic AI problems, as well as the models and the algorithms devised to address them. The course is divided in three main parts. In the first one, we will study algorithms for the resolution of informed search problems in state space, online search with/without the presence of an opponent, and constraint satisfiability problems. The second part will focus on the reasoning and decisional processes in the case of uncertainty. We will discuss ways to represent knowledge, including incomplete and uncertain knowledge of the real world. We will then focus on the logical reasoning over the acquired knowledge, using probabilities, and on using these reasoning methods and models to decide what to do. In the last part of the course, we will introduce distributed decision problems. Particularly, we will address game theory approaches for non-cooperative interaction decision problems, and the enforcement of such methods to concrete challenges.

PROGRAM

Introduction

- History of AI

Problem Solving

- Types of problems: single-state, multiple-states, context-dependent, heuristic problems, and exploration problems.
- Representation structures and problem formalization for the first three typologies.
- Uninformed Search Strategies:
 - Breadth-first search;
 - Depth-first search, and Iterative deepening depth-first search;
 - Comparing uninformed search strategies.
- Informed (Heuristic) Search Strategies:
 - Greedy best-first, A* and admissible heuristics;
 - Consistent heuristics;
 - Iterative Deepening A*;
 - Local search algorithms: Hill-climbing search, Local beam search, Simulated Annealing and Genetic.
- Multi-agent games: zero-sum games as problems to solve;
 - Utility functions and minimax strategy;

- Alpha beta pruning.

- Constraint Satisfaction Problems

- CSP as a search problem;
- Backtracking for CSP, Variable and value ordering;
- Constraint propagation: Forward Checking and AC3;
- The Structure of Problems;
- Local Search for CSPs.

Uncertain knowledge and reasoning

- Uncertainty representation:

- Introduction to decision theory, Basic Probability Notation, Atomic Events, Unconditional Probability, Inference Using Full Joint Distributions, Conditional Probability;
- Independence and conditional independence, Bayes' Rule and Its Use;

- Bayesian networks:

- The Semantics of Bayesian Networks;
- Conditional independence relations in Bayesian networks;
- Exact Inference in Bayesian Networks: Inference by enumeration, variable elimination algorithm.

- Reasoning in conditions of uncertainty:

- Actions and expected utility, Decision Networks and representing a decision problem with a decision network, Value of Information;
- Sequential Decision Problems, the utilities of states and Bellman equation for utilities, Value Iteration, Policy Iteration.

Multi-Agent systems and Game Theory

- Introduction to multi-agent systems;

- Introduction to game theory:

- Games in normal form and solutions
- Pure and mixed strategies; Nash equilibrium, Pareto optimality, social welfare;
- Zero-sum games and extended representation;
- Computation of Equilibria and Dominant Strategies;
- Zero-sum games solutions, minmax and alfabetta.
- Repeated games.

- Computational social choice:

- Voting mechanisms: social choice and social welfare functions;
- Arrow property and theorem;

- Muller-Satterthwaite theorem;
- Ranking functions and PageRank;
- Voting strategies and design mechanism (dominant strategies and Bayes Nash);
- Allocating scarce resources:
 - Auction mechanisms, individual-item auctions (English auction, Dutch auction, Japanese auction, Vickrey);
 - Equilibrium and dominant strategies in auction mechanisms;
- Reaching agreements:
 - Negotiation, domains;
 - Negotiation in task-oriented domains;
 - Theoretic games approaches, alternate bidding protocols;
 - Heuristic approaches, monotone concession protocol, Zeuthen strategy;
 - Multi-issue negotiation.

TEACHING MODALITIES

Lectures. Exercises.

TEACHING MATERIALS

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☒	Numeric	☒
Other						

SCHEDA DELL' INSEGNAMENTO Multimedia information systems

TITOLO INSEGNAMENTO IN INGLESE Multimedia information systems

Docente: Walter Balzano

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso tratta i principali modelli e tecniche per la gestione dei dati e dei sistemi informativi multimediali. Particolari riferimenti sono relativi ai meccanismi di storing, ricerca e browsing per contenuto su database multimediali, relazione tra database multimediali ed il Web. Sono inoltre introdotte strutture di sistemi informativi advanced quale quella dei sistemi informativi geografici con particolare attenzione ai sistemi di localizzazione quali GPS, Fingerprinting ed INS.

PROGRAMMA

Il corso è suddiviso in due parti. Prima parte: definizioni e classificazioni dei Media e dei Multimedia. Gestione di dati multimediali audio/video, dalla digitalizzazione alla consultazione degli stessi con particolari riferimenti ai concetti di Storing, Digital Signal Processing, Compressione e Streaming. Seconda parte: Sistemi Multimediali Digitali, Distribuiti ed Interattivi. Valutazioni di complessità, controllo ed adattamento. Presentazione ed interfacce utente. Un particolare interesse è rivolto ai Sistemi Informativi Multimediali orientati al Web e di quelli orientati ai Sistemi di Localizzazione InDoor ed outDoor, il Global Positioning System, Fingerprinting ed Inertial Measurement System.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Guojun Lu, Multimedia Database Management Systems, Artech House.

Blanken, de Vries, Blok, Feng, Multimedia Retrieval, Springer.

Ahmed El-Rabbany, Introduction to GPS, Artech House.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Multimedia information systems

Teacher: Walter Balzano

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

Propaedeutic courses: None

OBJECTIVES

The course deals with the main models and techniques for data management and multimedia information systems. Particular references are related to the storage, search and navigation mechanisms for contents on multimedia databases, relationships between multimedia databases and the Web. Furthermore, advanced information systems structures such as geographic information systems are introduced with particular attention to location systems such as GPS, fingerprints and INS.

PROGRAM

The course structured in two sections. First section: definitions and classifications of media and multimedia. Management of multimedia audio / video data, from digitization to recovery with particular reference to the concepts of storage, digital signal processing, compression and streaming. Second section: digital multimedia systems, distributed and interactive. Evaluations, control and adaptation of complexity. Presentation and user interfaces. Particular interest is focused on web-oriented multimedia information systems and InDoor and outDoor localization systems, the global positioning system, the fingerprint and inertial measurement system.

TEACHING MODALITIES

Lessons, exercises

TEACHING MATERIALS

Guojun Lu, Multimedia Database Management Systems, Artech House.
Blanken, de Vries, Blok, Feng, Multimedia Retrieval, Springer.
Ahmed El-Rabbany, Introduction to GPS, Artech House.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Natural language processing

TITOLO INSEGNAMENTO IN INGLESE Natural Language Processing

Docente: Francesco Cutugno

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso di Elaborazione del Linguaggio Naturale si pone come obiettivo il trasferimento agli studenti di conoscenze nel settore della linguistica computazionale. A questo obiettivo si giunge attraverso la presentazione di aspetti di linguistica, di elaborazione dei segnali, di vari tipi di metodologie stocastiche per il pattern recognition applicati alla voce umana, di algoritmi e tecniche per produrre voci artificiali, di strumenti e metodi per il trattamento dei corpora linguistici. Si illustrano inoltre le tecnologie, prevalentemente basate sulle moderne metodologie del machine learning e della intelligenza artificiale, per sviluppare applicazioni finalizzate al trattamento dei testi e ai sistemi automatici di gestione del dialogo uomo macchina.

PROGRAMMA

Dopo una introduzione alla Linguistica Generale (cenni) e una descrizione delle principali sottodiscipline in cui la Linguistica si articola, il corso procede articolandosi in due rami principali: elaborazione del testo scritto ed elaborazione della lingua parlata. **Gli argomenti per elaborazione del testo sono:** Espressioni Regolari, normalizzazione dei testi, Minimum Edit Distance, N-grammi, Part-of-Speech Tagging. Parsing sintattico a costituenti e a dipendenze, Word sense disambiguation e Wordnet, Rappresentazione di lessici e semantica in database a grafi.

Gli argomenti per la elaborazione della lingua parlata sono: Cenni di fonetica acustica e di fonologia articolatoria, l'uso del software PRAAT e del suo linguaggio di scripting, Hidden Markov Models, Sintesi Vocale, Riconoscimento Automatico del Parlato, Sistemi di dialogo con agenti artificiali, Spoken Language Understanding.

MODALITA' DIDATTICHE

Lezioni frontali, esercitazioni in aula

MATERIALE DIDATTICO

Jurafsky D. & Martin J.H. Speech and Language Processing, Prentice Hall 3rd ed.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera		Esercizi numerici	
Altro	Progetto di un tool per l'Elaborazione di testi e/o segnali vocali					

TEACHING SUMMARY OF THE COURSE Natural language processing

Teacher: Francesco Cutugno

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

Propaedeutic courses: None

OBJECTIVES

The Natural Language Processing class is finalized to give students solid bases and strong knowledge in the field of Computational Linguistics. To pursue this aim the course deals with the following arguments: general linguistics signal processing, speech processing, and pattern matching techniques, ASR and TTS, corpus linguistics. Moreover, lessons will offer know-how in machine learning and artificial intelligence applied to text processing and automatic human-machine dialogue management.

PROGRAM

Basis of General Linguistics, discipline articulation and terminology.

Text processing: Regular Expression, text normalization, N-grams, POS-Tagging, Syntactic Parsing, word sense disambiguation, word embedding, wordnet, lexicons and graph databases.

Speech Processing: Acoustic phonetics, PRAAT software, Hidden Markov Models, Speech synthesis, Speech recognition, Spoken dialogue systems, spoken language understanding.

TEACHING MODALITIES

Lezioni frontali, esercitazioni in aula

TEACHING MATERIALS

Jurafsky D. & Martin J.H. Speech and Language Processing, Prentice Hall 3rd ed.

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Design of a tool.					

SCHEDA DELL'INSEGNAMENTO DI Operating systems for mobile, cloud and IoT

TITOLO INSEGNAMENTO IN INGLESE Operating systems for mobile, cloud and IoT

Docente: Silvio Barra

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Il corso di Sistemi Operativi per dispositivi mobile, cloud e IoT ha una duplice finalità. In primo luogo, esso intende completare lo studio dei sistemi operativi tradizionali e real-time, approfondendo la gestione della memoria, dei processi, dello scheduling e della sicurezza in un sistema operativo specifico, quale Linux. D'altro canto, approfondisce i medesimi aspetti nel contesto dei sistemi operativi mobile, Cloud e IoT. In particolar modo, si analizza la struttura dei principali mobile OS, ovvero iOS e Android. Il corso approfondisce i principi alla base della virtualizzazione, con particolare attenzione alla piattaforma VMware, le tecnologie GRID e Cloud.

PROGRAMMA

Sistema Operativo Linux:

- Gestione della memoria: segmentazione, paginazione, Buddy-system.
- Gestione dei processi: creazione, sospensione, terminazione, scheduling.
- Sistemi operativi hard real-time: task periodici e aperiodici, scheduling, esempi di sistemi reali.

Sistemi Operativo mobile:

- il mercato e gli utenti, i Personal Digital Assistant, gli Smartphone.
- Mac iOS;
- Android OS;
- I webOS.

La virtualizzazione:

- I principi di base.
- L'architettura di VMware.
- La gestione della memoria in VMware.
- La gestione della CPU in VMware.

I sistemi Cloud:

- La tecnologia Grid.
- La tecnologia Cloud.
- Livelli di Cloud.
- Cloud Pervasivi.

Sistemi per IoT:

- Le architetture.
- Il mercato e i dispositivi COTS.
- Interconnessione e comunicazione fra dispositivi.
- Privacy e sicurezza.

MODALITA' DIDATTICHE

Lezioni frontali.

MATERIALE DIDATTICO

Daniel P. Bovet, Marco Cesati. Understanding the Linux Kernel (3° Edition).O'Reilly. 2005. Dispense del corso.

MODALITA' D'ESAME:

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Operating systems for mobile, cloud and IoT

Teacher: Silvio Barra

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

This course has a dual purpose. At first, it aims to integrate knowledge of traditional and real-time operating systems by deepening some internals like memory management structures and protocols, the process management and scheduling, and security requirements in a specific operating system, such as Linux. On the other hand, this course explores the same aspects in the context of mobile, Cloud and IoT operating systems. In particular, the attention focuses on the architecture of the main mobile OS, i.e. iOS and Android. The course also introduces the principles behind virtualization (with particular attention to the VMware platform) and GRID systems and then focuses on Cloud technologies.

PROGRAM

Linux Operating System:

- Memory management: segmentation, paging, Buddy-system.
- Process management: creation, suspension, termination, scheduling.
- Hard real-time operating systems: periodic and aperiodic tasks, scheduling, examples of real systems.

Mobile Operating Systems:

- the market and users, Personal Digital Assistants, Smartphones.
- Mac iOS;
- Android OS;
- The webOS.

Virtualization:

- The basic principles.
- The architecture of VMware.
- Memory management in VMware.
- CPU management in VMware.

Cloud systems:

- Grid technology.
- Cloud technology.
- Cloud levels.
- Pervasive Clouds.

IoT systems:

- The architectures.
- The market and COTS devices.
- Interconnection and communication between devices.
- Privacy and security.

TEACHING MODALITIES

Lectures.

TEACHING MATERIAL

Daniel P. Bovet, Marco Cesati. Understanding the Linux Kernel (3° Edition). O'Reilly. 2005. Lecture notes.

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☒	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Operations Research



TITOLO INSEGNAMENTO IN INGLESE Operations Research

Docente: Paola Festa

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
MATH-06/A	6	X			X		

Insegnamenti propedeutici previsti: Algoritmi e strutture dati I

OBIETTIVI FORMATIVI

L'insegnamento si prefigge quale obiettivo principale l'introduzione degli studenti all'uso dei modelli di programmazione matematica ed in particolare ai modelli di ottimizzazione lineare (sia continui che a variabili intere) ed alle loro applicazioni nei campi della logistica, dei servizi e della produzione industriale. L'impostazione metodologica del Corso, inoltre, punta al conseguimento dei seguenti ulteriori obiettivi intermedi:

- capacità di formalizzazione dei modelli di ottimizzazione per problemi di logistica, organizzazione, pianificazione, scheduling, trasporto, flusso su reti e problemi su grafi ed alberi;
- conoscenza della teoria e dei metodi di ottimizzazione lineare continua, di ottimizzazione lineare discreta e di ottimizzazione su grafi, alberi e reti di flusso;
- capacità di utilizzazione dei modelli matematici dei classici problemi di ottimizzazione e dei relativi algoritmi di risoluzione nei campi della Pianificazione della Produzione, della Localizzazione, della Gestione delle Scorte e della Logistica.

PROGRAMMA

Problemi di Programmazione Lineare e Metodo del Simplex. Definizione e classificazione dei problemi di ottimizzazione e dei problemi di decisione e classificazione dei relativi metodi risolutivi (metodi esatti, metodi di approssimazione e metodi euristici). Programmazione Lineare (PL): il Metodo del Simplex. Problemi di Programmazione Lineare Intera (1 credito) Metodi esatti per la risoluzione dei problemi di Programmazione Lineare Intera (Branch & Bound; piani di taglio; programmazione dinamica). Esempi di problemi di PLI con matrice dei vincoli uni-modulare: il problema del trasporto ed il problema dell'assegnamento. Problemi dello Zaino. Un algoritmo Branch and Bound per il problema dello Zaino 0/1; un algoritmo greedy per il problema dello Zaino Frazionario; due algoritmi di Programmazione Dinamica per il problema dello Zaino 0/1. Problemi di Ottimizzazione su grafi ed alberi: Vertex Cover ed Albero di Copertura Minimo. Il problema del Vertex Cover: un algoritmo 2-approssimato per il problema del Vertex Cover. Il problema dell'albero di copertura di un grafo a costo minimo (MST): l'algoritmo di Kruskal. Problemi di Ottimizzazione su grafi ed alberi: Problemi di Cammino Minimo. Cammini in un grafo orientato: il problema della raggiungibilità (visita in ampiezza; visita in profondità). Il problema dei cammini minimi: l'algoritmo di Dijkstra; l'algoritmo di Floyd e Warshall. Problemi di Ottimizzazione su grafi ed alberi: Pianificazione di un Progetto e Problema del Massimo Flusso. Pianificazione di un progetto: il Metodo CPM. Problemi di flusso su reti: il problema del massimo flusso; teorema max-flow min-cut; algoritmo di Ford-Fulkerson.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Appunti e Dispense del Corso

D. Bertsimas e J.N. Tsitsiklis, Introduction to Linear Optimization, Belmont - Massachusetts (USA), Dynamic Ideas and Athena Scientific, 2008

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☐
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☒	Esercizi numerici	☒
Altro						

TEACHING SUMMARY OF THE COURSE Operations Research



Teacher: Paola Festa

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
MATH-06/A	6	X			X		

Propaedeutic courses: Algorithms and Data Structures I**OBJECTIVES**

The main objective of the course is the introduction of the students to the use of mathematical programming models and in particular to both linear and nonlinear optimization models (with both continuous and integer variables) and their applications in real world fields, including control, communications, logistics, services, and industrial production.

The methodological approach of the course also aims at achieving the following additional milestones:

- knowledge of the theory and mathematical models of classic optimization problems such as network flows and graph problems and problems arising in management of stock and logistics, organization, planning, scheduling, transportation, decision systems design, control, communications, power systems, and resource allocation problems;
- knowledge of the fundamentals of linear and discrete optimization theory and methods. Topics include unconstrained and constrained optimization, linear and quadratic programming;
- ability to use the mathematical models of the classic optimization problems and the related solving algorithms in the fields of Production Planning, Localization, Inventory Management and Logistics.

PROGRAM

Linear Programming Problems and the Simplex Method. Definition and classification of optimization problems and decision problems and classification of the relative solution methods (exact methods, approximation methods and heuristic methods). Linear Programming (LP): the Simplex Method. Integer Linear Programming Problems. Exact methods for solving Integer Linear Programming problems (Branch & Bound; cutting plans; dynamic programming). Examples of ILP problems with uni-modular constraint matrix: the Transportation Problem and the Assignment Problem. Knapsack problems. A Branch and Bound algorithm for the 0/1 Knapsack Problem; a greedy algorithm for the Fractional Knapsack Problem; two Dynamic Programming algorithms for the 0/1 Knapsack Problem. Optimization problems on graphs and trees: Vertex Cover and Minimum Spanning Tree. The Vertex Cover Problem: a 2-approximation algorithm for the Vertex Cover Problem. The Minimum Spanning Tree of a graph (MST): the Kruskal algorithm. Optimization Problems on Graphs and Trees: Minimum Path Problems. Pay in an oriented graph: the problem of reachability (visit in breadth; visit in depth). Shortest Path Problems: Dijkstra's algorithm; the algorithm of Floyd and Warshall. Optimization Problems on Graphs and Trees: Planning a Project and Maximum Flow Problem. Planning a project: the CPM method. Flow problems on networks: the maximum flow problem; max-flow min-cut theorem; Ford-Fulkerson algorithm.

TEACHING MODALITIES

Frontal lessons. Exercises.

TEACHING MATERIALS

Notes provided by the lecturer.

D. Bertsimas e J.N. Tsitsiklis, Introduction to Linear Optimization, Belmont - Massachusetts (USA), Dynamic Ideas and Athena Scientific, 2008

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☒	Numeric	☒
Other						

SCHEDA DELL' INSEGNAMENTO DI Parallel High-Performance Computing: tools, algorithms, and software

TITOLO INSEGNAMENTO IN INGLESE Parallel High Performance Computing: tools, algorithms, and software

Docente: Valeria Mele

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Fornire idee di base, metodologie, strumenti per lo sviluppo di algoritmi e software in ambiente di calcolo ad alte prestazioni paralleli e/o distribuiti. Parte integrante del corso è l'attività di laboratorio.

PROGRAMMA

Influenza della memoria gerarchica sulle prestazioni del software. Algoritmi a blocchi per l'algebra lineare e relativo software. Modelli di programmazione per gli ambienti di calcolo parallelo/distribuito ad alte prestazioni più diffusi: cluster computing, multicore computing, network computing e GPU computing. Esempi significativi di software parallelo ad alte prestazioni: ScaLAPACK, MAGMA. Cenni alle nuove tipologie di ambienti di calcolo: grid computing e cloud computing. Modelli di programmazione ibridi. Strumenti Python per il Calcolo Parallelo e ad Alte Prestazioni.

MODALITA' DIDATTICHE

lezioni frontali e esercitazioni in laboratorio

MATERIALE DIDATTICO

D. Kirk, W.W. Hwu, Programming Massively Parallel Processor second ed., Morgan Kaufman
J. Dongarra et al., Sourcebook of parallel computing, Morgan Kaufman
A. Murli, Lezioni di Calcolo Parallelo, Liguori ed.
Risorse web, lucidi delle lezioni, materiale fornito dal docente

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono (*)	A risposta multipla		A risposta libera		Esercizi numerici	
Altro						

TEACHING SUMMARY OF THE COURSE Parallel High-Performance Computing: tools, algorithms, and software

Teacher: Valeria Mele

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

OBJECTIVES

The main aim of the course is to provide basic ideas, methodologies, and tools for the development of algorithms and software in a parallel and distributed high-performance computing environment. An essential part of the course is the laboratory activity.

PROGRAM

Impact of hierarchical memory on software performance. Block algorithms for linear algebra and related software. Programming models for the most common high-performance parallel / distributed computing environments: cluster computing, multicore computing, network computing, and GPU computing. Significant examples of parallel high-performance software: ScaLAPACK, MAGMA. Notes on new types of computing environments: grid computing and cloud computing. Hybrid programming models. Python tools for Parallel Calculation and High Performance.

TEACHING MODALITIES

Lectures and laboratory exercises

TEACHING MATERIALS

D. Kirk, W.W. Hwu, Programming Massively Parallel Processor second ed., Morgan Kaufman

J. Dongarra et al., Sourcebook of parallel computing, Morgan Kaufman

A. Murli, Lezioni di Calcolo Parallelo, Liguori ed.

Web resources, slides of the lesson, material supplied by the teacher

EXAM

The exam is given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other						

SCHEDA DELL' INSEGNAMENTO DI Parallel and Distributed Computing



TITOLO INSEGNAMENTO IN INGLESE Parallel and Distributed Computing

Docente: Marco Lapegna

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X			X		

Insegnamenti propedeutici previsti: Nessuno.

OBIETTIVI FORMATIVI

Fornire idee di base, metodologie, strumenti software per lo sviluppo di algoritmi in ambiente di calcolo parallelo e/o distribuito. Parte integrante del corso è l'attività di laboratorio.

PROGRAMMA

Concetto di “parallelismo” e di “alte prestazioni”. I supercomputer. Classificazione e principali caratteristiche funzionali delle architetture parallele (classificazione di Flynn, rivista e aggiornata). Parametri di valutazione delle prestazioni degli algoritmi paralleli. I parametri classici di SpeedUp ed Efficiency. Metodologie per lo sviluppo di algoritmi paralleli e loro dipendenza dall'architettura. Esempi di progettazione e implementazione di algoritmi su architetture di tipo MIMD distributed memory (uso di message programming; la libreria MPI) e di tipo MIMD shared memory (l'esempio dei multicore; condivisione di memoria; la libreria OpenMP). Parametri di valutazione e scalabilità degli algoritmi paralleli. SpeedUp scalato ed Efficiency scalata. Legge di Amdahl; legge di Ware generalizzata. Legge di Gustafson. Il bilanciamento del Carico. Vari esempi (e relativa implementazione) di algoritmi paralleli per il calcolo matriciale. Cenni alle GPU e alla libreria CUDA. La documentazione del software parallelo: linee guida.

MODALITA' DIDATTICHE

Lezioni frontali e attività in laboratorio; sviluppo di piccoli progetti durante il corso.

MATERIALE DIDATTICO

- A. Murli: Lezioni di Calcolo Parallelo, Liguori Editore.
- B. Chapman, G. Jost, R. van der Pas: Using OpenMP. Portable Shared Memory Parallel Programming”, The MIT Press, 2007
- un manuale di MPI
- un manuale di OpenMp
- Slides delle lezioni; altro materiale fornito dal docente

MODALITA' D'ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono	A risposta multipla		A risposta libera	X	Esercizi numerici	X
Altro	Realizzazione e consegna di piccoli progetti durante il corso; OPPURE prova pratica di Laboratorio					

TEACHING SUMMARY OF THE COURSE Parallel and Distributed Computing



Teacher: Marco Lapegna

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X			X		

Propaedeutic courses: None

OBJECTIVES

The main aim of the course is to provide basic ideas, methodologies, and tools for the development of algorithms and software in a parallel and/or distributed computing environment. An essential part of the course is the laboratory activity.

PROGRAM

Concepts of “parallelism” and “high performances”. Supercomputers. Parallel architectures classification and their main functional characteristics (revised and updated Flynn’s classification). Parallel algorithms performance evaluation parameters. The “SpeedUp” and “Efficiency” classical parameters. Parallel algorithms design and development methodologies; the influence of the architecture. Design and implementation of parallel algorithms on MIMD distributed memory architectures (the message passing: the MPI library) as well as on MIMD shared memory ones (multicores; memory sharing; the Open MP library). Performance evaluation and scalability of parallel algorithms: scaled SpeedUp and scaled Efficiency. Amdahl's law; generalized Ware’s law. Gustafson law. The workload balancing. Several examples (and implementation) of matrix computation parallel algorithms. GPU and the CUDA library. Parallel software documentation: guidelines.

TEACHING MODALITIES

frontal lessons and laboratory exercises and practice; (very) small projects development

TEACHING MATERIALS

- A. Murli: Lezioni di Calcolo Parallelo, Liguori Editore.
- B. Chapman, G. Jost, R. van der Pas: Using OpenMP. Portable Shared Memory Parallel Programming”, The MIT Press, 2007
- MPI manual
- OpenMp manual
- Web resources; slides of the lessons; other materials supplied by the teacher

EXAM

The exam is given in form	Written and oral	☒	Written only	☐	Oral only	☐
In case of written exams, the tests are	Multiple choice	☐	Open answer	☒	Numeric	☒
Other	Small projects during the course; OR a laboratory test					

SCHEDA DELL' INSEGNAMENTO DI Scientific Computing

TITOLO INSEGNAMENTO IN INGLESE Scientific Computing

Docente: Eleonora Messina

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
MATH-05/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Cosa sono i metodi numerici e come si usano per risolvere i problemi di ingegneria e delle scienze? L'obiettivo primario di questo corso è quello di fornire agli studenti una risposta concreta a queste domande attraverso esempi e casi di studio. Obiettivi più specifici sono: identificare approcci numerici adatti a particolari problemi, proporre strategie di risoluzione e comprendere le tecniche implementative più efficaci per la risoluzione del problema stesso. I temi principali trattati nel corso sono: ricerca di radici di equazioni non lineari, approssimazione ed interpolazione, integrazione e derivazione, equazioni differenziali. In tutti i casi si analizzano metodologie per il problem solving e la relativa implementazione in Matlab.

Risultati di apprendimento previsti:

- comprendere la relazione tra gli aspetti teorici ed i problemi pratici che nascono nella risoluzione di un problema;
- analizzare e confrontare metodi numerici sulla base delle diverse caratteristiche dei problemi da risolvere;
- sviluppare algoritmi numerici ed usare le funzioni Matlab;
- risolvere i problemi computazionali attraverso la progettazione di algoritmi strutturati;
- interpretare i risultati computazionali anche in relazione all'accuratezza ed all'efficienza dei metodi.

PROGRAMMA

Modelli matematici, metodi numerici e problem solving. Programmazione con Matlab: operazioni di base, M-file, programmazione strutturata, metodologie per il problem solving. Errori di round-off e di troncamento. Radici di equazioni non lineari: metodi grafici, bisezione e falsa posizione, Newton-Raphson, secanti e metodi ibridi. Approssimazione ed interpolazione: regressione lineare, interpolazione polinomiale, spline ed interpolazione a tratti. Integrazione numerica: formule di Newton-Cotes, regola dei trapezi, di Simpson, quadratura adattiva. Derivata numerica: formule di ordine alto. Problemi ai valori iniziali: il metodo di Eulero e sue modifiche. Problemi ai limiti: metodi alle differenze finite.

MODALITA' DIDATTICHE

Lezioni, attività di problem solving, esercitazioni laboratorio, esercizi e progetti.

MATERIALE DIDATTICO

- R.L. Burden, J.D. Faires, Numerical Analysis, Brooks/Cole Pub Co, 2015;
- S. C. Chapra, Applied numerical methods with MATLAB for engineers and scientists 3. international ed.: Boston: McGraw-Hill Higher Education, 2012.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Esercizi individuali di programmazione e progetti con relazione scritta.					

TEACHING SUMMARY OF THE COURSE Scientific Computing

Teacher: Eleonora Messina

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
MATH-05/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

What are numerical methods and how do they relate to engineering and scientific problem solving? The objective of the course is to provide students with concrete answers to these questions through worked examples and case studies. More specific objectives are: identify numerical approaches for methodic scientific problems, propose strategies for solving them and understand how numerical methods work to generate a solution through implementation on a digital computer. The major area searched in this course will be root finding, curve fitting, integration, differentiation, differential equations. In all cases methodologies for problem solving and their Matlab implementations are analysed.

Learning outcomes:

- understand how theory connects to practical issues in problem solving;
- analyse and compare numerical methods on the basis of different application problems to be solved;
- develop numerical algorithms and use Matlab functions;
- solve computational problems in a structured way and implement in Matlab;
- interpret computational results also in relation to efficiency and accuracy.

PROGRAM

Mathematical modeling, numerical methods and problem solving. Programming with Matlab: fundamentals, M-file, structured programming, problem solving methodologies. Roundoff and truncation errors. Root finding: graphical methods, bisection and false position, Newton-Raphson, Secant and hybrid methods. Curve fitting: linear regression, polynomial interpolation, splines and piecewise interpolation. Numerical integration: Newton-Cotes formulas, Trapezoidal rule, Simpson rules, adaptive quadrature. Numerical differentiation: high accuracy differentiation formulas. Initial value problems: Euler's method and improvements. Boundary value problems: finite difference methods.

TEACHING MODALITIES

Lectures, problem solving classes, computer laboratory sessions, assignments and projects.

TEACHING MATERIALS

- R.L. Burden, J.D. Faires, Numerical Analysis, Brooks/Cole Pub Co, 2015;
- S. C. Chapra, **Applied numerical methods with MATLAB for engineers and scientists** 3. international ed.: Boston: McGraw-Hill Higher Education, 2012.

EXAM

The exam will be given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Individual programming exercises and projects with written report.					

SCHEDA DELL' INSEGNAMENTO DI Security and Privacy

TITOLO INSEGNAMENTO IN INGLESE Security and Privacy

Docente: Piero Andrea Bonatti

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6		X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso si propone di fornire allo studente una panoramica il più possibile completa delle problematiche relative alla sicurezza e alla privacy informatiche e delle tecniche per affrontarle. Pertanto, il corso spazia dai modelli di sicurezza alle tecniche crittografiche, agli standard emergenti relativi alla sicurezza e alla privacy in ambito informatico, coprendo sia aspetti schiettamente tecnologici che alcuni fondamenti teorici. Il corso comprende sia approcci ormai ben assestati che alcune direzioni innovative che promettono di essere assorbite nella tecnologia e negli standard più comuni.

PROGRAMMA

Introduzione su terminologia di base e servizi di sicurezza; autenticazione, controllo degli accessi, audit. Controllo degli accessi: politiche, modelli e meccanismi. Modelli discrezionale e mandatorio, modello basato su ruoli, politiche di integrità, politiche amministrative e problema della revoca, separazione dei privilegi, e autorizzazioni per classi e gerarchie. Meccanismi di sicurezza comuni (in DBMS, Web servers, Firewalls, Java) tra cui database multilivello e polistanziamento. Sicurezza nelle reti: scanning, spoofing, session hijacking e denial-of-service; vulnerabilità dei protocolli TCP/IP e contromisure; virus, trojan horses e rootkits; firewalls e loro ACL; limiti dei firewall e vulnerabilità applicative. Problematiche peculiari delle reti wireless. Privacy: problematiche, e standards (P3P); problemi di inferenza in database statistici; soluzioni per macrodati e microdati. Approcci *game theoretic* alla privacy. Crittografia simmetrica e asimmetrica: breve storia e metodi moderni: da DES a RSA (con richiami di algebra e dimostrazioni di correttezza). Tecniche ed infrastrutture crittografiche per la sicurezza delle reti: PKI, PGP, SSL, SSH, Kerberos.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

Stallings: *Sicurezza delle reti*. Mc Graw Hill
Articoli, dispense e slides forniti dal docente

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☒	Solo scritta	☐	Solo orale	☐
In caso di prova scritta i quesiti sono	A risposta multipla	☐	A risposta libera	☒	Esercizi numerici	☐
Altro						

TEACHING SUMMARY OF THE COURSE Security and Privacy



Teacher: Piero Andrea Bonatti

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6		X		X		

Propaedeutic courses: None

OBJECTIVES

This course presents a broad introduction to the many different aspects of computer security and privacy, and related techniques. Therefore, topics range from security models to cryptographic techniques, network security, malware, confidential data publishing, and security and privacy standards, just to name a few. Accordingly, the course includes both technological and theoretical aspects.

PROGRAM

Introduction on basic terminology and security services: access control, authentication and audit. Accesso control: policies, models, and mechanisms. Discretionary and mandatory models, role-based access control, integrity policies, administrative policies and revocation problems, separation of duties, authorization inheritance. Common security mechanisms in DBMS, Web servers, Firewalls, Java, including multilevel databases and polynstantiation. Network security: scanning, spoofing, session hijacking and denial-of-service; vulnerabilities of TCP/IP and countermeasures; virus, trojan horses and rootkits; firewalls and their ACL; firewall limitations and application vulnerabilities. Peculiarities of wireless networks. Privacy: issues and standards standards (P3P); inference problems in statistical databases; solutions for macrodata and microdata. Game-theoretic approaches to privacy. Symmetric and asymmetric cryptography. DES, AES, and RSA. Frameworks, protocols and tools for network security: PKI, PGP, SSL, SSH, Kerberos.

TEACHING MODALITIES

Lezioni frontali. Esercitazioni.

TEACHING MATERIALS

Stallings: *Sicurezza delle reti*. Mc Graw Hill

Articles and slides.

EXAM

The exam is given in form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice		Open answer	X	Numeric	
Other						

SCHEDA DELL' INSEGNAMENTO DI Social, ethical and psychological issues in Artificial Intelligence

TITOLO INSEGNAMENTO IN INGLESE Social, ethical and psychological issues in Artificial Intelligence

Docente: Lucio Franzese

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X	X			X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Lo studente acquisirà la capacità di identificare e analizzare problematiche sociali, etiche, legali ed economiche collegate allo sviluppo e all'uso responsabile delle tecnologie e dei sistemi IC, con particolare riferimento alle tecnologie e ai sistemi dell'intelligenza artificiale e della robotica.

Lo studente acquisirà inoltre la capacità di applicare concetti, metodologie e tecniche di base dell'informatica allo studio scientifico dei processi di elaborazione dell'informazione dei sistemi biologici.

Lo studente acquisirà infine conoscenze di base sulla storia dell'IA e consapevolezza delle principali svolte metodologiche nello sviluppo di questo e di altri ambiti disciplinari affini.

PROGRAMMA

- Introduzione alle tematiche ELSE (Etiche, Legali e Socio-Economiche) relative alle tecnologie e ai sistemi IC, con particolare attenzione alle tecnologie e ai sistemi dell'IA e della robotica. Analisi di casi di studio esemplari per la valutazione di impatto ELSE di tali tecnologie e sistemi.
- Introduzione alle metodologie per lo studio scientifico di sistemi biologici cognitivi e adattativi mediante simulazioni informatiche e robotiche. Analisi di casi di studio esemplari attinti dalle neuroscienze cognitive, da neuropsicologia, psicologia cognitiva, e modellistica del comportamento animale.
- Introduzione alla storia dell'IA e all'evoluzione dei suoi obiettivi e delle sue metodologie.

MODALITA' DIDATTICHE

Lezioni frontali. Esercitazioni.

MATERIALE DIDATTICO

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	X	Solo scritta		Solo orale	
In caso di prova scritta i quesiti sono (*)	A risposta multipla		A risposta libera		Esercizi numerici	
Altro	Saggi brevi (1000 parole) in itinere. Tesina finale di circa 15-30 pagine					

TEACHING SUMMARY OF THE COURSE Social, ethical, and psychological issues in Artificial Intelligence

Teacher: Lucio Franzese

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X	X			X	

Propaedeutic courses: None

OBJECTIVES

Students will develop basic capabilities for identifying and analysing social, ethical, legal, and economic issues concerning the responsible design, implementation, and use of IC technologies and systems, with a special emphasis on AI and robotic technologies and systems.

Students will additionally develop the capability to apply basic computer science concepts, methodologies and techniques to the scientific investigation and modelling of information processing in both humans and other biological systems.

Students will also acquire knowledge of historical landmarks and awareness of major turning points in the development of AI.

PROGRAM

- Introduction to ELSE (Ethical, Legal and Socio-Economic) issues concerning IC, AI and robotics technologies and systems. Exemplary case-studies for the evaluation of their ELSE impact.
- Introduction to methodologies for the computational and robotic modelling of adaptive behaviours and cognitive capabilities of biological systems. Exemplary case-studies for computational and robotic modelling taken from cognitive psychology, neuropsychology, cognitive neuroscience, and animal behaviour studies.
- Introduction to the historical development of AI, its evolving methodologies and goals.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

EXAM

The exam will be given in form	Written and oral	X	Written only		Oral only	
In case of written exams, the tests are	Multiple choice		Open answer		Numeric	
Other	Short essays (ca 1000 words) as homework assignments. A typically 15-30 pages long final paper.					

SCHEDA DELL' INSEGNAMENTO DI Software Project Management and Evolution



TITOLO INSEGNAMENTO IN INGLESE Software Project Management and Evolution

Docente: Sergio Di Martino

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Italiano
INFO-01/A	6	X				X	

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

L'obiettivo del corso è di fornire conoscenze avanzate su metodi e tecniche dell'ingegneria del software per la gestione di progetti complessi, anche in uno scenario di evoluzione di sistemi preesistenti. Gli studenti saranno in grado di sviluppare, mantenere e gestire l'evoluzione di sistemi software con architetture complesse e distribuite, di scegliere le metodologie e tecnologie più adatte a risolvere un determinato problema, di scegliere tra diversi modelli di rappresentazione del software quello più adatto a rappresentare un determinato aspetto del sistema. Gli studenti saranno inoltre in grado di produrre documenti software in accordo a standard di progetto.

In particolare, gli studenti riceveranno nozioni di: processi moderni di ingegneria del software e delle relative fasi, con particolare riferimento alle Metodologie Agili; metodi, standard e strumenti per la gestione della qualità del prodotto e del processo; stili e notazioni per la modellazione di architetture software e in particolare architetture distribuite, cloud-based e orientate ai microservizi, a tecniche avanzate di testing, verifica e manutenzione del software.

PROGRAMMA

Software Maintenance e Reverse Engineering: Manutenzione del Software – Reengineering, refactoring, restructuring, reverse engineering– Tools per la Build Automation – Tools per il Controllo di Versione

Metodologie di sviluppo Software: Metodi Agili, SCRUM, Continuous Integration, Continuous Deployment. Behaviour-Driven Development

Architetture Software: Cloud, IaaS, PaaS, SaaS. Microservice-oriented Architectures. Docker.

Qualità del Software: Modelli e Standard di Qualità del Software – Metriche di Qualità del Software – Metriche di Taglia del Software - Function Points.

Testing: Livelli e Modelli di Testing – Black Box Testing – White Box Testing – Integration Testing – User Interface Testing – Testing Automation – Metriche e Strumenti per la Coverage di Test.

MODALITA' DIDATTICHE

Lezioni Frontali.

MATERIALE DIDATTICO

Ian Sommerville, Ingegneria del Software, Pearson Addison Wesley;

Roger Pressman, Principi di Ingegneria del Software, Mc Graw Hill.

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale	☐	Solo scritta	☐	Solo orale	☒
In caso di prova scritta i quesiti sono (*)	A risposta multipla	☐	A risposta libera	☐	Esercizi numerici	☐
Altro	Progetto o Tesina facoltativi					

TEACHING SUMMARY OF THE COURSE Software Project Management and Evolution



Teacher: Sergio Di Martino

SSD	CFU	Year			Semester		Language
		I	II		I	II	Italian
INFO-01/A	6	X				X	

Propaedeutic courses: None

OBJECTIVES

The aim of the course is to provide advanced knowledge on Software Engineering Methods and Techniques for managing complex software projects, also in a scenario of software evolution. Students will be able to: design, develop and maintain complex software systems based also on distributed architecture; identify the most suitable methodologies and technologies to solve a problem; choose between different models of representation of the software. Students will also be able to produce documents in accordance with quality standard.

In particular, students will be introduced to: modern software engineering processes and related phases, especially to Agile Methodologies; methods, standards and tools for product and process quality management; tools and notations for designing/modeling software architectures, including distributed, cloud-based and microservice-oriented architectures, advanced techniques of software testing, verification and maintenance.

PROGRAM

Software Maintenance and Reverse Engineering: Maintenance – Reengineering, refactoring, restructuring, reverse engineering– Tools for Build Automation – Tools for Version Control

Software Methodologies: Agile Methods, SCRUM, Continuous Integration, Continuous Deployment. Behaviour-Driven Development

Software Architectures: Cloud, IaaS, PaaS, SaaS. Microservice-oriented Architectures. Docker.

Software Quality: Software Quality Standards and Models – Software Quality Metrics – Software Size Metrics - Function Points.

Testing: Testing Levels and Models – Black Box Testing – White Box Testing – Integration Testing – User Interface Testing – Testing Automation – Coverage Metrics and Tools.

TEACHING MODALITIES

Lectures.

TEACHING MATERIALS

Ian Sommerville, Ingegneria del Software, Pearson Addison Wesley;
Roger Pressman, Principi di Ingegneria del Software, Mc Graw Hill.

EXAM

The exam will be given in form	Written and oral	Written only	Oral only	X
In case of written exams, the tests are	Multiple choice	Open answer	Numeric	
Other	Optional project or essay			

SCHEDA DELL' INSEGNAMENTO DI Generative AI

TITOLO INSEGNAMENTO IN INGLESE Generative AI

Docente: Anna Corazza

SSD	CFU	Anno di corso			Semestre		Lingua
		I	II		I	II	Inglese
INFO-01/A	6	X	X		X		

Insegnamenti propedeutici previsti: Nessuno

OBIETTIVI FORMATIVI

Il corso mira a fornire metodologie e tecniche per comprendere e affrontare i problemi dell'Intelligenza Artificiale Generativa. Gli studenti acquisiranno concetti chiave, modelli e architetture che costituiscono l'IA Generativa, partendo dalle reti neurali ricorrenti, dal meccanismo di attenzione e dai transformer. Le applicazioni includeranno non solo l'elaborazione del linguaggio naturale, ma anche il dominio multimodale. Altri approcci considerati includeranno le Reti Generative Avversarie (GAN), gli Autoencoder, gli Autoencoder Variazionali (VAE) e i Modelli di Diffusione. Gli studenti acquisiranno competenza nelle tecniche avanzate di deep learning necessarie per progettare e addestrare modelli generativi, comprendendo come funzionano, come possono essere adattati a compiti specifici e valutati, considerando anche la loro implementazione nei framework più utilizzati.

PROGRAMMA

Per il programma dettagliato, consultare la scheda dell'insegnamento su docenti.unina oppure contattare la docente.

MODALITA' DIDATTICHE

Lezioni frontali in presenza. Esercitazioni.

MATERIALE DIDATTICO

MODALITA' DI ESAME

L'esame si articola in prova	Scritta e orale		Solo scritta		Solo orale	X
In caso di prova scritta i quesiti sono (*)	A risposta multipla		A risposta libera		Esercizi numerici	
Altro	Discussione di un progetto + orale					

TEACHING SUMMARY OF THE COURSE Generative AI

Teacher: Anna Corazza

SSD	CFU	Year			Semester		Language
		I	II		I	II	English
INFO-01/A	6	X	X		X		

Propaedeutic courses: None

OBJECTIVES

The course aims to provide methodologies and techniques to understand and address the problems of Generative Artificial Intelligence.

Students will acquire key concepts, models, and architectures that constitute Generative AI, starting from recursive neural networks, the attention mechanism and transformers. Applications will include not only natural language processing but also the multimodal domain. Other considered approaches will include Generative Adversarial Networks (GANs), Autoencoders, Variational Autoencoders (VAEs), and Diffusion Models. Students will gain proficiency in advanced deep learning techniques required to design and train generative models, understanding how they work, how they can be adapted to specific tasks and assessed by also considering their implementation in the most widely used frameworks.

PROGRAM

For the detailed program, consult the course sheet on docenti.unina.it or contact the teacher.

TEACHING MODALITIES

Lectures (in-person).

TEACHING MATERIALS

EXAM

The exam will be given in form	Written and oral	☐	Written only	☐	Oral only	☒
In case of written exams, the tests are	Multiple choice	☐	Open answer	☐	Numeric	☐
Other	Discussion of a project + oral exam.					