

Allegato B2

Quadro degli obiettivi formativi specifici e delle propedeuticità

Corso di Laurea Magistrale in Informatica

Rau, art. 12

Insegnamento	Settore Scientifico Disciplin.	Obiettivi formativi specifici (ITA)	Specific educational objectives (ENG)	Propedeuticità obbligatorie
Advanced Data Science	INF/01	<p>Gran parte delle attività economiche moderne non potrebbero aver luogo senza i dati, che quindi rappresentano fattori essenziali della produzione come i macchinari e le persone. L'uso efficace dei dati, la loro analisi e visualizzazione allo scopo di estrarne informazione e conoscenza, ha il potenziale per trasformare le economie, offrendo una nuova ondata di crescita della produttività e maggior tempo libero per le persone. I dati possono svolgere un ruolo economico significativo a vantaggio non solo del commercio privato, ma anche delle economie nazionali e dei loro cittadini, in particolare nel settore dell'assistenza sanitaria, dell'amministrazione pubblica, e nella soluzione di problemi globali del nostro pianeta.</p> <p>Nel corso affronteremo l'analisi e la visualizzazione di dati nel linguaggio R, in particolare gli argomenti trattati includono:</p> <ol style="list-style-type: none">1. introduzione a R2. introduzione al flusso dell'analisi dei dati: importazione, normalizzazione, trasformazione, visualizzazione, modellizzazione e comunicazione3. la scienza delle reti: centralità	<p>Much of modern economic activity could not take place without data, which are therefore essential factors of production such as machinery and people. The effective use of data, its analysis and visualization for the purpose of extracting information and knowledge, has the potential to transform economies, offering a new wave of productivity growth and more leisure time for people. Data can play a significant economic role to the benefit not only of private trade, but also of national economies and their citizens, particularly in health care, public administration, and in solving global problems on our planet.</p> <p>In the course we will address the analysis and visualization of data in the R language, in particular the topics covered include:</p> <ol style="list-style-type: none">1. introduction to R2. introduction to the data analysis flow: import, normalization, transformation, visualisation, modelling and communication3. network science:	

		<p>e potere, similarità, comunità, resilienza, distanze e piccolo mondo, leggi di potenza e reti ad invarianza di scala 4. analisi del testo: frequenza delle parole e dei documenti, analisi dei sentimenti, n-grammi e co-apparizione di termini, topic modeling 5. blockchain</p> <p># Capacità relative alle discipline</p> <p>## Conoscenza e capacità di comprensione: lo studente dovrà aver acquisito le conoscenze e le capacità necessarie per analizzare e visualizzare dati strutturati e non strutturati e per comunicare i risultati dell'analisi</p> <p>## Conoscenza e capacità di comprensione applicate: lo studente dovrà essere in grado di usare l'ambiente R e in particolare le librerie tidyverse e tidytext per l'analisi e la visualizzazione dei dati</p> <p># Capacità trasversali/soft skills</p> <p>## Autonomia di giudizio: lo studente dovrà essere in grado di interpretare i risultati dell'analisi dei dati e trarre conclusioni attinenti al dominio del discorso</p> <p>## Abilità comunicative: lo studente dovrà essere in grado di comunicare in modo efficace e accattivante i risultati dell'analisi dei dati</p> <p>## Capacità di apprendere: lo studente dovrà dimostrare di aver appreso la tecnica di analisi</p>	<p>centrality and power, similarity, community, resilience, distances and small worlds, power laws and scale-free networks 4. text analysis: frequency of words and documents, sentiment analysis, n-grams and co-appearance of terms, topic modeling 5. blockchain</p> <p># Discipline-related skills</p> <p>## Knowledge and understanding: the student must have acquired the necessary knowledge and skills to analyse and visualise structured and unstructured data and to communicate the results of the analysis.</p> <p>## Applied knowledge and understanding: the student should be able to use the R environment and in particular the tidyverse and tidytext libraries for data analysis and visualization.</p> <p># Cross skills / soft skills</p> <p>## Autonomy of judgement: the student should be able to interpret the results of the data analysis and draw conclusions relevant to the domain of discourse</p> <p>## Communicative skills: the student should be able to communicate the results of the data analysis in an effective and engaging way.</p>	
--	--	---	--	--

		dei dati sia strutturati che in formato testo e di saperla utilizzare in un contesto qualsiasi	## Learning skills: the student will have to demonstrate that they have learned the technique of analysis of data, both structured and in text format, and to know how to use it in any context	
Advanced database systems for big data	INF/01	<p>Il corso ha lo scopo di far acquisire allo studente una conoscenza approfondita di tematiche avanzate di gestione dei dati nel paradigma relazionale (tecniche avanzate di processamento e ottimizzazione di interrogazioni, progettazione fisica e basi di dati distribuite) nonché paradigmi e linguaggi alternativi per la gestione dei dati (basi di dati in XML).</p> <p>Il corso mira inoltre a fornire competenze tecniche e strumenti inerenti alla gestione e analisi di grandi quantitativi di dati, con particolare attenzione alle tecniche di data warehousing e data mining, nonché a metodi e strumenti specifici per i "big data". Tra gli argomenti trattati ci saranno il paradigma MapReduce e le blockchain e loro applicazioni.</p> <p>Alla fine del corso lo studente sarà in grado di valutare le prestazioni delle basi di dati e attuare misure di correzione, avrà assimilato i concetti e le metodologie fondamentali per la configurazione di sistemi per basi di dati distribuite e per l'analisi di piccoli e grandi quantitativi di dati.</p>	<p>The overall aim of the course is to acquire an in-depth knowledge on advanced topics in data management within the relational paradigm (advanced query processing and optimization techniques, physical design, and distributed database systems), as well as alternative data models and languages (XML databases).</p> <p>In addition, the course aims at providing competences about techniques and tools for big data management and analysis. A special attention will be given to data warehousing, data mining, and other methods and tools specific for big data. A number of key topics will be addressed, ranging from the MapReduce paradigm to blockchain and its applications.</p> <p>At the end of the course, the student will be able to evaluate and tune the performance of a database, will have learned the concepts and methodologies for the configuration of distributed databases, and for the analysis of small and big data.</p>	

	<p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione</p> <ul style="list-style-type: none"> - Architetture per sistemi paralleli e sistemi distribuiti di basi di dati. - Partizionamento e replicazione dei dati in sistemi paralleli e/o distribuiti. - Tecniche di processamento e ottimizzazione di interrogazioni sia in un contesto centralizzato sia in ambito distribuito. - Modelli di dato alternativi a quello relazionale per la gestione di dati semi-strutturati e non strutturati. - Caratteristiche dei sistemi di nuova generazione (NoSQL e NewSQL). <p>1.2. Conoscenza e capacità di comprensione applicate</p> <ul style="list-style-type: none"> - Tecniche e strumenti per l'analisi e la visualizzazione di piccoli e grandi quantitativi di dati (es.: R e RStudio). - Tecniche di ottimizzazione delle prestazioni di un sistema di gestione dei dati relazionale. - Processamento dei dati con sistemi non relazionali (es.: XML e MapReduce). <p>Capacità trasversali/soft skills</p>	<p>Sector-specific skills</p> <p>1.1. Knowledge and understanding</p> <ul style="list-style-type: none"> - Parallel and distributed database system architectures. - Data partitioning and replication in parallel and distributed systems. - Centralized and distributed query processing and optimization. - Alternative data model (with respect to the relational paradigm) for semi-structured and unstructured data. - Features of new generation (NoSQL, NewSQL) systems. <p>1.2. Applying knowledge and understanding</p> <ul style="list-style-type: none"> - Techniques and tools for small and big data analysis and visualization (e.g., R and RStudio). - Optimization techniques for performance improvement in relational systems. - Data processing in non-relational systems (e.g. XML and MapReduce). 	
--	--	---	--

	<p>2.1. Autonomia di giudizio</p> <ul style="list-style-type: none"> - Saper scegliere le tecniche corrette e gli strumenti appropriati per condurre analisi di dati. - Interpretare i risultati di un'analisi e trarre conclusioni concrete rilevanti al dominio di interesse. - Capacità di determinare l'architettura (centralizzata, parallela, distribuita, relazionale o meno) più adatta a risolvere un problema di gestione dei dati. - Porre in essere le strategie più opportune per migliorare le prestazioni delle interrogazioni. <p>2.2. Abilità comunicative</p> <ul style="list-style-type: none"> - Saper comunicare attraverso il lessico tecnico dei sistemi di basi di dati. - Saper comunicare usando la terminologia dei sistemi paralleli e distribuiti. - Saper comunicare con i soggetti direttamente e indirettamente coinvolti nel processo di progettazione e sviluppo di una basi di dati, nonché con i soggetti utilizzatori (es.: saper comunicare in maniera chiara i risultati di un'analisi). 	<p>Cross-sectoral skills/soft skills</p> <p>2.1. Making judgments</p> <ul style="list-style-type: none"> - Choose the correct techniques and the appropriate tools to carry out data analyses. - Interpret the experimental results of the analysis and draw effective conclusions relevant to the domain of discourse. - Determine the most suitable (centralized, parallel, distributed, relational or non-relational) architecture for a specific data management problem. - Implement the best strategies to improve the query performance. <p>2.2. Communication skills</p> <ul style="list-style-type: none"> - Communicate using the technical lexicon of database systems. - Communicate using the terminology of parallel and distributed systems. - Communicate with the (technical and non-technical) stakeholders involved in the process of design, implementation, and use of a database system (e.g., 	
--	--	---	--

		<p>2.3. Capacità di apprendimento</p> <ul style="list-style-type: none"> - Saper ottimizzare un sistema di gestione dei dati, anche parallelo o distribuito. - Saper scegliere un campione di dati abbastanza ricco per poter condurre un'analisi da cui estrarre informazioni significative, nonché trarre e <ul style="list-style-type: none"> • comunicare le conclusioni derivanti. 	<p>communicate effectively the results of the analysis).</p> <p>2.3. Learning skills</p> <ul style="list-style-type: none"> - Learn to optimize a (possibly parallel or distributed) data management system. - Learn to choose a sufficiently rich row data set, to analyze the data to <ul style="list-style-type: none"> • extract meaningful information, to draw and to communicate conclusions. 	
Algoritmi avanzati	INF/01	<p>Indice:</p> <ul style="list-style-type: none"> • Algoritmica su stringhe <ul style="list-style-type: none"> ○ <i>Algoritmi e tecniche di base per la ricerca esatta su stringhe.</i> ○ <i>Algoritmo Z, Knth-Morris-Pratt, Boyer-Moore</i> ○ <i>Le strutture dati per la ricerca: Suffix Tries, Suffix Trees, Suffix Arrays</i> ○ <i>Distanze</i> ○ <i>La ricerca approssimata: programmazione dinamica</i> ○ <i>Algoritmi e tecniche di base per la ricerca approssimata su stringhe</i> ○ <i>Smith-Watermann, Landau-Vishkin</i> • Algoritmi randomizzati <ul style="list-style-type: none"> ○ <i>Randomness e Algoritmica</i> ○ <i>Algoritmi e tecniche di base</i> ○ <i>Algoritmo di Rabin-Karp</i> ○ <i>Universal Hashing</i> 	<p>Index:</p> <ul style="list-style-type: none"> • Algorithmics on strings <ul style="list-style-type: none"> ○ <i>Algorithms and basic techniques for exact pattern matching.</i> ○ <i>Algorithms Z, Knth-Morris-Pratt, Boyer-Moore</i> ○ <i>Data structures for string searching: Suffix Tries, Suffix Trees, Suffix Arrays</i> ○ <i>Distances</i> ○ <i>Approximate pattern matching: dynamic programming</i> ○ <i>Algorithms and basic techniques for approximate pattern matching</i> ○ <i>Smith-Watermann, Landau-Vishkin</i> • Randomized algorithms 	

	<ul style="list-style-type: none"> • Algoritmi paralleli <ul style="list-style-type: none"> ○ <i>Algoritmi paralleli e distribuiti: modelli di calcolo</i> ○ <i>Algoritmi e tecniche di base</i> ○ <i>Algoritmo BoxSort</i> <p>Lo/la studente/essa dovrà:</p> <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione Conoscere le tecniche algoritmiche di base per l'analisi e l'indicizzazione di collezioni di stringhe, per l'algoritmica randomizzata, per l'algoritmica su architetture parallele o distribuite. Conoscere la collezione dei principali algoritmi e strutture dati utilizzati nei tre campi studiati.</p> <p>1.2 Capacità di applicare conoscenza e comprensione Saper implementare disegnare ed analizzare algoritmi avanzati. Saper scegliere ed utilizzare strutture dati avanzate. Saper implementare algoritmi che siano in grado di sfruttare randomness e architetture non convenzionali</p> <p>Capacità trasversali / soft skills</p> <p>2.1 Autonomia di giudizio Saper scegliere architettura e strutture dati adeguate per la soluzione di problemi algoritmici su stringhe. Saper valutare le possibilità offerte dalla randomizzazione. Saper valutare l'utilizzo di idee algoritmiche per le più popolari architetture non convenzionali. Saper valutare la complessità e la implementabilità di uno strumento algoritmico non elementare.</p>	<ul style="list-style-type: none"> ○ <i>Randomness and Algorithmica</i> ○ <i>Algorithms and basic techniques</i> ○ <i>Rabin-Karp algorithm</i> ○ <i>Universal Hashing</i> <ul style="list-style-type: none"> • Parallel algorithms <ul style="list-style-type: none"> ○ <i>Parallel and distributed algorithms: computation models</i> ○ <i>Algorithms and basic techniques</i> ○ <i>BoxSort algorithm</i> <p>The student will be able to:</p> <p>Sector-specific skills</p> <p>1.1. Knowledge and understanding Understand the basic algorithmic techniques for analyzing and indexing strings collections, for random algorithm, for algorithmic on parallel or distributed architectures. Know the collection of the main algorithms and data structures used in the three fields studied.</p> <p>1.2 Applying knowledge and understanding Know how to deploy and analyze advanced algorithms. Know how to choose and use advanced data structures. Know how to implement algorithms that are able to exploit randomness and unconventional architectures</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements Know how to choose suitable architecture</p>	
--	--	--	--

		<p>2.2 Abilità comunicative. Essere in grado di comunicare e argomentare in relazione a scelte algoritmiche, strutture dati, tecniche di disegno algoritmico/implementazione, ed architetture. Saper valutare i trade-off più significativi durante le scelte e la valutazione di una soluzione algoritmica proposta.</p> <p>2.3 Capacità di apprendimento Saper valutare ed integrare nuove idee e tecniche algoritmiche/implementative, strutture dati e metodi di analisi della complessità computazionale.</p>	<p>and data structures for solving algorithmic problems on strings. Know how to evaluate the possibilities offered by randomization. Know how to use algorithmic ideas for the most popular unconventional architectures. Know how to evaluate the complexity and implementability of a non-elemental algorithmic tool.</p> <p>2.2 Communication skills Communicate and argue in relation to algorithmic choices, data structures, algorithmic design / implementation techniques, and architectures. Know how to evaluate the most significant trade-offs along the choices and the analysis of a proposed algorithmic solution.</p> <p>2.3 Learning skills Know how to evaluate and integrate new ideas and algorithmic / implementation techniques, data structures and analytical methods of computational complexity.</p>	
Algoritmi numerici e applicazioni	INF/01	<p>Il corso descrive algoritmi, metodi e tecniche computazionali ampiamente utilizzati nell'ambito del calcolo scientifico, sottolineandone l'importanza per le applicazioni. Opportune attività di laboratorio consentiranno l'analisi critica di alcuni metodi presentati. Le nozioni di base di algebra lineare verranno riviste.</p> <p>Capacità relative alle discipline</p> <p>1.1 Conoscenze e capacità di comprensione Valutare la complessità di un algoritmo numerico e collocarla nell'ambito dei limiti noti per il problema. Saper distinguere fra le diverse</p>	<p>The course focuses on algorithms, methods and computational techniques of widespread use in scientific computing, underlining their importance in applications. Suitable laboratory activities will enable the critical analysis of some of the presented methods. The needed numerical linear algebra will be revised.</p> <p>Sector-specific skills</p> <p>1.1 Knowledge and understanding Evaluate the complexity of a numerical algorithm in connection with the known bounds for the problem. Recognize the</p>	

		<p>strutture che una matrice può presentare. Conoscere i principali aspetti teorici della trasformata di Fourier, tra i quali il legame con la convoluzione.</p> <p>1.2 Conoscenze e capacità di comprensione applicate</p> <p>Utilizzare un algoritmo appropriato per risolvere un sistema lineare. Saper utilizzare la trasformata veloce di Fourier per l'analisi di dati mono e bidimensionali.</p> <p>Capacità trasversali/Soft Skills</p> <p>2.1 Autonomia di giudizio</p> <p>Valutare criticamente un algoritmo alla luce dei risultati noti di complessità, paragonandolo ad altri algoritmi per lo stesso problema.</p> <p>2.2 Abilità comunicative</p> <p>Descrivere le principali caratteristiche di un algoritmo numerico contestualizzandolo ed utilizzando un'appropriata terminologia.</p> <p>2.3 Capacità di apprendimento</p> <p>Leggere libri o articoli in riviste specializzate per l'approfondimento autonomo circa i temi affrontati nel corso.</p>	<p>different structures that a matrix can have. Know the most relevant aspects of the theory of the Fourier transform and its relation with convolution.</p> <p>1.2 Applying knowledge and understanding</p> <p>Solve a linear system by using a suitable algorithm. Use fast Fourier transform for analyzing mono and two-dimensional data.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making Judgments</p> <p>Evaluate an algorithm in the light of known complexity results, comparing it to other algorithms for the same problem.</p> <p>2.2 Communications skills</p> <p>Describe the main features of a numerical algorithm by placing it in the right perspective and using appropriate terminology.</p> <p>2.3 Learning skills</p> <p>Read books or articles in specialized journals in order to deepen the topics studied in the course.</p>	
Analisi e verifica mediante interpretazione astratta	INF/01	<p>Primo obiettivo formativo del corso è l'introduzione dei fondamenti della manipolazione formale dei sistemi software mediante Interpretazione Astratta. Inoltre ci si prefigge la familiarizzazione con le tecniche di analisi statica e trasformazione di programmi basati sulla semantica che permettono, tra l'altro, lo sviluppo di strumenti automatici per l'ottimizzazione e, soprattutto, per la verifica, il debugging, e la correzione automatica del software.</p>	<p>The first objective of the course is to introduce the fundamentals of the formal manipulation of software systems through Abstract Interpretation. In addition it aims to familiarize with static analysis and program transformation techniques based on semantics that allow, among other things, the development of automatic tools for optimization and, above all, for verification, debugging, and automatic correction of software.</p>	

		<p>Il corso presenterà le nozioni base dell'Interpretazione Astratta con cui si andranno a costruire alcuni esempi di metodi di manipolazione di programmi, sia di analisi che di debugging che di verifica.</p> <p>Data l'odierna vastità di utilizzi dell'Interpretazione Astratta in moltissime aree dell'informatica, nella parte finale del corso si andranno ad approfondire alcune applicazioni specifiche in base alle preferenze manifestate dagli studenti.</p> <p>Le tecnologie illustrate nel corso saranno presentate in modo rigoroso, sia per quanto riguarda gli aspetti teorici che implementativi. Verranno presentati dei sistemi esistenti su cui poter eseguire esercitazioni pratiche.</p> <p>Capacità relative alle discipline</p> <p><i>1.1 Conoscenza e capacità di comprensione:</i> Lo studente conosce i principali concetti della Interpretazione Astratta; le caratteristiche fondamentali degli strumenti di analisi statica del software; i principali concetti relativi alle tecniche di verifica automatica del software che si possono costruire mediante Interpretazione Astratta.</p> <p><i>1.2 Conoscenza e capacità di comprensione applicate:</i> Lo studente impara le tecniche base dell'Interpretazione Astratta con cui può costruire dei prototipi di strumenti di manipolazione di programmi basati sulla semantica. Inoltre impara a capire quali siano le potenzialità e i limiti di questo approccio. Grazie a vari esercizi assegnati durante il corso e successivamente corretti lo studente consolida</p>	<p>The course will present the basic notions of Abstract Interpretation, which will be used to build examples of program manipulation methods, either of analysis, of debugging and of verification.</p> <p>Given today's vast amount of uses of Abstract Interpretation in many areas of computer science, in the final part of the course we will go into some specific applications basing on the preferences expressed by the students.</p> <p>The technologies illustrated in the course will be rigorously presented, both as regards the theoretical that the implementative aspects.</p> <p>Sector-specific skills</p> <p><i>1.1 Knowledge and understanding</i> The student knows the main concepts of Abstract Interpretation; The basic features of software static analysis tools; The main concepts related to automated software verification techniques that can be constructed by using Abstract Interpretation.</p> <p><i>1.2 Applying knowledge and understanding</i> Students learn the basic techniques of Abstract Interpretation with which he can build prototypes of semantics-based program manipulation tools. They also learn what the potentialities and limitations of this approach are. Through various exercises assigned during the course and subsequently corrected, students consolidate the theoretical knowledge presented during lectures.</p>	
--	--	--	--	--

		<p>le conoscenze teoriche presentate durante le lezioni frontali.</p> <p>Capacità trasversali/soft skills</p> <p><i>2.1 Autonomia di giudizio:</i> Questo insegnamento migliora la capacità dello studente di identificare problematiche comuni e relative soluzioni nell'ambito della progettazione e dello sviluppo di strumenti di manipolazione di programmi basati sulla semantica. Inoltre, imparando a ragionare in termini di possibili approssimazioni decidibili di problemi che sono invece indecidibili in forma esatta, apprende dei concetti avanzati e delle tecniche che potrebbero avere notevole impatto anche in situazioni applicative completamente diverse.</p> <p><i>2.2 Abilità comunicative:</i> Lo studente impara l'esatto significato dei termini usati nell'analisi e nella verifica dei sistemi software.</p> <p><i>2.3 Capacità di apprendere:</i> Gli esercizi assegnati durante il corso e la loro correzione offrono allo studente gli strumenti necessari a valutare il proprio grado di apprendimento.</p>	<p>Cross-sectoral skills/soft skills</p> <p><i>2.1 Making judgements</i> This teaching improves the students' ability to identify common issues and their solutions in the design and development of semantic-based program manipulation tools. In addition, by learning to think in terms of possible decidable approximations of problems that are undecidable in exact form, they learn advanced concepts and techniques that could have a significant impact even in completely different application situations.</p> <p><i>2.2 Communication skills</i> The student learns the exact meaning of the terms used in the fields of analysis and verification of software systems.</p> <p><i>2.3 Learning skills</i> The exercises assigned during the course and their correction offer students the tools they need to evaluate their level of learning.</p>	
Auditory and Tactile Interactions	INF/01	<p>Al termine del corso gli studenti saranno in possesso di alcune semplici metodologie per elaborare il suono e per sintetizzare vibrazioni, con particolare riferimento alla sintesi in tempo reale. In particolare: avranno appreso i fondamenti della psicologia della percezione uditiva e tattile; conosceranno la natura di un segnale e di un sistema a tempo continuo e discreto; sapranno riconoscere e valutare lo spettro di un segnale a tempo discreto;</p>	<p>After completing the course, students will master some simple sound and vibration processing methods, with specific regard to real time synthesis. In particular, they will:</p> <ul style="list-style-type: none"> - have learned the fundamentals of psychology of auditory and tactile perception; - know the nature of continuous-time and discrete signals and systems; - recognize and evaluate the spectrum of a 	

	<p>saranno a conoscenza delle tecniche tradizionali per la sintesi del suono; avranno nozioni sufficienti per valutare e progettare semplici sistemi software e hardware per la sintesi di suoni e vibrazioni; avranno compreso i principi base del funzionamento del software per la sintesi del suono in tempo reale, su cui avranno elaborato degli esempi di realizzazioni di modelli per la sintesi del suono e di vibrazioni.</p> <p>Capacità relative alle discipline: Conoscenza e comprensione Gli studenti acquisiscono durante il Corso di Auditory and Tactile Interactions conoscenza di base dell'elaborazione di suono e vibrazioni nelle sue componenti fondamentali percettive, matematiche, e pratiche. Le stesse conoscenze sono applicate alla sintesi in tempo reale al calcolatore.</p> <p><i>Capacità di applicare conoscenza e comprensione</i> Attraverso la soluzione di un numero sufficiente di esercizi, gli studenti comprendono la difficoltà di progettare e successivamente realizzare architetture software e hardware di generazione di suono e vibrazioni per applicazioni in tempo reale.</p> <p>Capacità trasversali /soft skills <i>Autonomia di giudizio</i> In virtù del radicamento dell'elaborazione del suono all'interno del campo più vasto della teoria dei segnali e della teoria dei sistemi, lo studente al termine del corso avrà maturato un'autonomia di giudizio di tipo più profondo relativamente alla bontà di progettazione e alle prestazioni dei costituenti fondamentali del</p>	<p>discrete-time signal; - be aware of conventional techniques for sound synthesis; - have sufficient knowledge to evaluate and design simple software and hardware systems for the synthesis of sounds and vibrations; - have understood the basic principles of the operation of the software for the synthesis of the sound in real time, on which they will elaborate examples of realizations of models for the synthesis of the sound and vibration.</p> <p>The expected learning outcomes are, as reflected in the following Dublin Descriptors.</p> <p>Sector-specific skills <i>Knowledge and understanding:</i> Students acquire basic knowledge about sound and vibrations in their fundamental components of perception, mathematics, and operation. The same knowledge are applied to real-time synthesis on the computer.</p> <p><i>Applying knowledge and understanding:</i> Through the solution of a sufficient number of exercises, the students understand the difficulty of designing and subsequently realize software architectures and hardware for the generation of sound and vibrations in real-time applications.</p> <p>Cross-sectoral skills/soft skills <i>Making judgments:</i></p>	
--	--	--	--

		<p>software per l'elaborazione dei segnali in tempo reale.</p> <p><i>Abilità comunicative</i></p> <p>Al termine del corso lo studente avrà cognizione dei più importanti risultati legati alla percezione audio-tattile dei segnali, e contemporaneamente darà un significato alla terminologia legata alle tecniche e tecnologie per la resa di segnali uditivi e tattili, molto adoperata anche nell'ingegneria della comunicazione: decibel, spettro, banda, risposta in frequenza solo per citare alcuni termini. Conseguentemente, lo studente avrà future possibilità di far interagire assieme le diverse professionalità, tipicamente in possesso di abilità creative o tecniche, che operano nel settore della realizzazione della componente non visuale delle interfacce.</p> <p><i>Capacità di apprendimento</i></p> <p>Sulla base dei concetti appresi lo studente potrà successivamente approfondire una molteplicità di aspetti legati alla psicofisica della percezione uditiva e tattile, al design d'interfacce audio-tattili, e alla progettazione di software per la resa di suono e vibrazioni.</p>	<p>Due to sound processing as part of the broader signals and systems theory field, the student at the end of the course will have acquired a deeper autonomy in judging the design and performance quality of the real-time signal processing software fundamental components.</p> <p><i>Communication skills:</i></p> <p>At the end of the course students will have knowledge of the most important results related to the audio-tactile perception of the signals, and simultaneously give a meaning to the terminology related to the techniques and technologies for the rendering of auditory and tactile signals, used a lot also in communication engineering: decibels, spectrum, bandwidth, frequency response just to mention a few terms. Consequently, the student will have future opportunities to interact with diverse professionals, typically possessing creative or technical skills, who are active in the realization of non-visual interface component.</p> <p>- <i>Learning skills:</i></p> <p>Based on the learned concepts, students will be able to explore a variety of issues in the psychophysics of auditory and tactile perception, the design of audio-tactile interfaces, and the design of software for rendering of sound and vibration.</p>	
Complessità e teoria dell'informazione	INF/01	<p>Indice:</p> <ul style="list-style-type: none"> • <i>Teoria della Complessità</i> 	<p>Index:</p> <ul style="list-style-type: none"> • <i>Complexity Theory</i> <ul style="list-style-type: none"> ○ <i>Time and Space complexity on Turing</i> 	

	<ul style="list-style-type: none"> ○ <i>Complessità in termini di Tempo e Spazio su Macchine di Turing e altri modelli classici</i> ○ <i>Relazioni tra le classi di complessità</i> ○ <i>Riduzioni, completezza e istanze di linguaggi nelle diverse classi</i> ○ <i>Modelli di calcolo non standard: DNA e Quantum Computing</i> ○ <i>Algoritmi su Grafi alla base della complessità computazionale: reachability, trace equivalence and bisimulation</i> <ul style="list-style-type: none"> ● Information Theory <ul style="list-style-type: none"> ○ <i>Concetti di base</i> ○ <i>Entropia e compressione dei dati</i> ○ <i>Mutua Informazione</i> ○ <i>Complessità di Kolmogorov</i> <p>Lo studente dovrà essere in grado di:</p> <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione Definire formalmente i modelli classici di calcolo e le classi di complessità in tempo e spazio. Presentare alcuni elementi di ogni classe di complessità studiata. Enunciare e dimostrare i risultati della teoria della complessità presentati durante il corso. Definire i modelli di calcolo DNA e Quantum e confrontarli con i modelli classici. Descrivere gli algoritmi su grafi. Definire le nozioni standard della teoria dell'informazione. Descrivere i risultati classici sulla compressione dei dati presentati durante il corso.</p> <p>1.2 Capacità di applicare conoscenza e comprensione Classificare i linguaggi in termini di complessità</p>	<p><i>Machines and other classical models</i></p> <ul style="list-style-type: none"> ○ <i>Relationships between complexity classes</i> ○ <i>Reductions, completeness and instances of languages in the different classes</i> ○ <i>Non standard computational models: DNA and Quantum Computing</i> ○ <i>Graph algorithms at the basis of computational complexity: reachability, trace equivalence and bisimulation</i> <ul style="list-style-type: none"> ● Information Theory <ul style="list-style-type: none"> ○ <i>Basic Concepts</i> ○ <i>Entropy and data compression</i> ○ <i>Mutual Information</i> ○ <i>Kolmogorov complexity</i> <p>The student should be able to:</p> <p>Sector-specific skills</p> <p>1.1. Knowledge and understanding Formally define the classical models of computation and the time/space complexity classes. Present some proper members of each studied complexity class. Present and prove the complexity theory results presented during the course. Define the DNA and Quantum models of computation and compare them with the</p>	
--	--	--	--

		<p>in tempo e spazio. Elaborare riduzioni tra i linguaggi. Definire e implementare algoritmi sui grafi per varianti dei problemi analizzati nel corso. Modellare e risolvere problemi semplici della teoria delle informazioni: compressione dei dati e codifica del canale.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio Stabilire se un problema può essere risolto in modo efficiente o no. Elaborare algoritmi efficienti per risolvere nuovi problemi. Introdurre vincoli per rendere un problema trattabile. Stimare le prestazioni di diversi sistemi di informazione e comunicazione.</p> <p>2.2 Abilità comunicative. Motivare le soluzioni proposte. Spiegare quali condizioni aggiuntive potrebbero contribuire a risolvere il problema in modo più efficiente. Giustificare le scelte del modello computazionale e delle strutture dati. Spiegare i metodi di codifica e di compressione e i limiti informativi.</p> <p>2.3 Capacità di apprendimento Trovare e sfruttare soluzioni esistenti su problemi correlati. Sfruttare nuovi strumenti per migliorare le complessità computazionali. Identificare e fornire soluzioni per problemi di teoria dell'informazione, codifica e comunicazione.</p>	<p>classical models. Describe algorithms on graphs. Define the standard notions of Information Theory. Present the classical results on data compression presented during the course.</p> <p>1.2 Applying knowledge and understanding Classify languages in terms of time and space complexity. Elaborate reductions between languages. Define and implement algorithms over graphs for variants of the problems analysed in the course. Model and solve simple problems of Information Theory: data compression and channel coding.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements Establish whether a problem can be efficiently solved or not. Elaborate efficient algorithms for solving new problems. Eventually introduce constraints to make a problem tractable. Estimate performances of different information and communication systems.</p> <p>2.2 Communication skills. Motivate the proposed solutions. Explain which additional conditions could help to solve the problem more efficiently. Justify the choices of the computational model and data structures. Explain coding and compression methods and information limits.</p>	
--	--	---	--	--

			<p>2.3 Learning skills Find and exploit existing solutions over related problems. Exploit new instruments for improving the computational complexities. Identify information, coding and communication problems/solutions.</p>	
Data & Techniques for E-Health	ING-INF/05	<p>Il corso di Informatica Medica si propone di introdurre gli studenti ai problemi specifici legati all'applicazione delle metodologie e tecnologie informatiche in ambito medico-clinico.</p> <p>Programma</p> <ul style="list-style-type: none"> • Il Sistema Sanitario in Italia • Caratteristiche dei dati e del ragionamento in Medicina • La cartella clinica elettronica <p>Il Fascicolo Sanitario Elettronico</p> <ul style="list-style-type: none"> • Standard di comunicazione in Sanità • Terminologie, classificazioni, ontologie • Sistemi basati sui protocolli (CDSS) • Elaborazione di biosegnali e bioimmagini • Telemedicina • eHealth • Casi di studio: sistemi, articoli scientifici <p>Lo/la studente/ssa dovrà:</p> <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione Conoscere le caratteristiche di base del dominio biomedico, sia dal punto di vista della rappresentazione ed utilizzo della conoscenza biomedica, che dell'organizzazione dei servizi sanitari e della loro informatizzazione. Conoscere funzioni e criticità dei sistemi di eHealth e Telemedicina.</p>	<p>The Medical Informatics course is aimed at introducing students to the problems related to introduction of Information and Communication Technologies and methodologies into the health care field.</p> <ul style="list-style-type: none"> ○ The Health care system in Italy ○ Data and reasoning in Medicine ○ Electronic Healthcare Records <ul style="list-style-type: none"> ○ An example: the Longitudinal Health Record ○ Medical informatics standards ○ Terminologies and classifications, ontologies ○ Protocol-based systems ○ Biosignal and bioimage processing ○ Telemedicine ○ eHealth ○ Examples: systems, scientific papers <p>At the end of the course the student is expected to:</p> <p>1.1. Knowledge and understanding know the basic features of the biomedical domain, from the point of view of the representation and exploitation of biomedical knowledge, as well as of the organization of healthcare services and their informatization.</p>	

		<p>1.2 Capacità di applicare conoscenza e comprensione Conoscere le caratteristiche di base del sistema sanitario nazionale. Saper identificare ed utilizzare gli standard e le tecnologie applicabili in un determinato contesto biomedico. Sapere quale conoscenza è rappresentabile con un'ontologia e quale no; conoscere le principali ontologie utilizzate in ambito biomedico ed i loro formati di rappresentazione. Sapere impostare un processo di elaborazione ed analisi di bioimmagini. Saper definire un sistema di telemedicina appropriato secondo le Linee Guida nazionali.</p> <p>Capacità trasversali / soft skills Lo/la studente/ssa dovrà:</p> <p>2.1 Autonomia di giudizio Saper esaminare ed inquadrare correttamente un determinato problema biomedico.</p> <p>2.2 Abilità comunicative. Avere acquisito un minimo di conoscenza di base del linguaggio utilizzato dai professionisti sanitari, in modo da essere in grado di interagire correttamente e comprenderne le esigenze alla base dei requisiti di un sistema di informatica medica. Essere in grado di motivare, a voce e per iscritto, le scelte concettuali effettuate nella progettazione di una soluzione informatica per un problema biomedico.</p> <p>2.3 Capacità di apprendimento Saper reperire ed utilizzare risorse informatiche e scientifiche per l'approfondimento autonomo delle tematiche studiate a lezione.</p>	<p>know functions and issues of eHealth and telemedicine systems.</p> <p>1.2 Applying knowledge and understanding To know basic features of the Italian Health care systems, as example of organization. To be able to identify and use standards and technologies suitable in a specific biomedical context. To know which kind of knowledge can be represented in an ontology and which is not; to know the main biomedical ontologies and their formats. To be able to design a biomedical image processing and analysis system. To be able to set up a telemedicine system according to national guidelines.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements Be able to study and correctly frame a specific biomedical problem.</p> <p>2.2 Communication skills. Acquire a basic knowledge of the language used by medical professionals, in order to be able to interact with them, understand their needs and translate them to some medical informatics system requirements. Be able to motivate the conceptual choices made when designing an informatic solution to a biomedical problem.</p> <p>2.3 Learning skills Be able to find and use informatic and</p>	
--	--	---	---	--

		Saper leggere ed approfondire un articolo scientifico in ambito biomedico.	scientific resources for an autonomous close examination of the course topics. Be able to read and delve into a biomedical scientific paper.	
Deep learning	ING-INF/05	<p>Il corso si propone di presentare nel dettaglio le architetture e gli algoritmi per la definizione, l'implementazione e l'addestramento di Reti Neurali Artificiali. Il Deep Learning, la cui traduzione letterale significa apprendimento profondo, è una sottocategoria del Machine Learning e indica quella branca dell'Intelligenza Artificiale che fa riferimento ai sistemi e agli algoritmi ispirati alla struttura e alla funzione del cervello chiamate Reti Neurali Artificiali.</p> <p>Recentemente grazie alla quantità crescente d'informazioni digitalmente disponibili il Machine Learning, e in particolare il Deep Learning, è diventato un importante settore dell'informatica con moltissime applicazioni in svariati settori tra cui la bioinformatica, la medicina, l'analisi automatica del testo, l'analisi automatica delle immagini e video, il riconoscimento del parlato, i sistemi di guida autonoma ecc. L'obiettivo principale del corso è quindi fornire allo studente le conoscenze fondamentali per risolvere problemi di apprendimento automatico attraverso una corretta formulazione del problema, una progettazione critica dell'architettura neurale da utilizzare e un'analisi sperimentale per valutare i risultati ottenuti. Per questo motivo il corso prevede lezioni in laboratorio al fine di consentire allo studente la sperimentazione diretta delle nozioni apprese su applicazioni reali.</p>	<p>The course aims to present architectures and algorithms in order to design, implement and train Artificial Neural Networks. Deep Learning is subfield of Machine Learning and it is a branch of Artificial Intelligence which refers to systems and algorithms inspired by the structure and function of the brain called Artificial Neural Networks.</p> <p>Recently, thanks to the increasing amount of data digitally available, Machine Learning, and Deep Learning in particular, has become an important field of computer science with several applications in different scientific areas such as bioinformatics, medicine, natural language processing, computer vision, speak recognition, autonomous driving systems, and so on.</p> <p>The course aims to enable students to acquire basic knowledge to solve machine learning tasks through a proper formulation of the problem, a critical choice of the neural network architecture and an experimental analysis of the obtained results. The course includes laboratory activities so that students can directly test on real applications concepts learned in class.</p>	

		<p>Capacità relative alle discipline</p> <p>1.1 Conoscenza e capacità di comprensione: durante il corso lo studente acquisirà conoscenze teoriche dei principali componenti utilizzati per lo sviluppo di una rete neurale e dei metodi di apprendimento delle stesse; inoltre, apprenderà le procedure formali per la valutazione e l'analisi dei risultati ottenuti.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: l'attività di laboratorio permetterà allo studente di consolidare le conoscenze teoriche presentate durante le lezioni frontali del docente attraverso il loro utilizzo in casi reali.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: lo studente acquisirà capacità teoriche e pratiche che gli consentiranno di sviluppare in autonomia algoritmi di Deep Learning e analizzarne criticamente i risultati prodotti.</p> <p>2.2 Abilità comunicative: lo studente imparerà la terminologia appropriata e sarà in grado di descrivere le principali caratteristiche delle architetture neurali e degli algoritmi di apprendimento presentati durante il corso.</p> <p>2.3 Capacità di apprendimento: il corso mira a fornire allo studente gli strumenti necessari per affrontare e risolvere autonomamente problemi inerenti all'apprendimento automatico attraverso lo sviluppo critico di architetture neurali.</p>	<p>Capacità relative alle discipline</p> <p>1.1 Knowledge and understanding: During the course, the student will learn basic knowledge of the main components used for neural network development and their learning strategies. He will also learn procedures for evaluating and analyzing the obtained results.</p> <p>1.2 Applied knowledge and understanding: the laboratory activity allows the student to consolidate the theoretical knowledge, presented during the lectures, through their use in real application cases.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomous assessments: the student will acquire theoretical and practical skills that will allow him to develop Deep Learning systems algorithms and to analyse critically the obtained results.</p> <p>2.2 Communication skills: the student will learn appropriate terminology and he will be able to present the main features of Deep Learning architectures and algorithms presented in the course.</p> <p>2.3 Learning skills: the course aims to provide students with the basic knowledge needed to understand and solve automatically machine learning problems using critically Deep Learning techniques.</p>	
Didattica dell'informatica	INF/01	Il corso si propone di stimolare una riflessione sulla natura dell'informatica in generale, anche in relazione alle altre discipline scientifiche e tecnologiche, e sul ruolo della programmazione	The course proposes a reflection on the scientific and methodological foundations of computer science, in general, and on	

		<p>in particolare, al fine di accrescere la consapevolezza in merito alla complessità dei processi mentali messi in gioco e delle conseguenti implicazioni per l'apprendimento. Verranno dapprima considerati alcuni contributi al dibattito sulla natura e sui fondamenti della disciplina, sia per metterne in luce la ricchezza e l'articolazione, sia per valutarne le implicazioni didattiche.</p> <p>Quindi si rifletterà sul ruolo della programmazione nell'informatica e sulla natura di questa attività, confrontando i diversi paradigmi di programmazione in relazione agli strumenti di astrazione che li caratterizzano. Infine, si discuteranno alcune tematiche oggetto di ricerca didattica nell'ambito della disciplina.</p> <p>Parte I - Fondamenti storico-epistemologici e didattica dell'Informatica</p> <ul style="list-style-type: none"> - Accezioni dell'informatica come strumento, tecnologia, disciplina. - Gli sviluppi della didattica dell'informatica nei programmi scolastici europei. - Modelli curricolari di riferimento: contenuti e approcci all'insegnamento. - Il dibattito filosofico sulla natura della disciplina: le anime matematico, ingegneristica e scientifica. - Altri punti di vista sulla natura dell'informatica. - Esempi di percorsi didattici che illustrano le diverse anime dell'informatica. - I concetti dell'informatica in una prospettiva storica. 	<p>computer programming as a key activity to appreciate the nature of computing. On this basis, it discusses the implications for the learning of computing topics and skills.</p> <p>The course will first discuss the nature, foundations and richness of computer science in connection with the related educational goals. Then, it will consider the learning of programming and the key role of abstraction in the programming tasks. Finally, a selection of research topics in computer science education will be discussed.</p> <p>Part I - Historical/epistemological foundations of computer science and computing education</p> <ul style="list-style-type: none"> - Computing as a tool vs. computing as a discipline. - Development of school computing education in Europe. - Curricular models: syllabi and teaching approaches. - The nature of computing: mathematical, scientific and engineering perspectives. - Other perspective about computing. - Examples of learning units in connection with the diversity of perspectives. - Computing concepts in a historical perspective. 	
--	--	---	---	--

		<p>Parte II - Natura e didattica della Programmazione</p> <ul style="list-style-type: none"> - Ruolo della programmazione e possibili approcci didattici nei modelli CC2001. - Introduzione alla programmazione: approcci imperative-first, functional-first e object-first. - I paradigmi di programmazione dal punto di vista didattico. - Ambienti e strumenti di supporto per la didattica della programmazione. - Le ricerche sull'apprendimento della programmazione. <p>Descrittori di Dublino</p> <p>Capacità relative alla disciplina</p> <p>Lo/la studente/ssa dovrà:</p> <p>1.1. Conoscenza e capacità di comprensione</p> <ul style="list-style-type: none"> - Conoscere le radici storiche ed epistemologiche dell'informatica; - Conoscere i principali modelli curriculari per l'insegnamento dell'informatica; - Conoscere alcuni strumenti di supporto all'apprendimento dell'informatica e della programmazione; - Conoscere alcune tecniche di valutazione dell'apprendimento della disciplina; - Conoscere le principali tematiche oggetto di ricerca nella didattica dell'informatica e della programmazione. <p>1.2 Capacità di applicare conoscenza e comprensione</p> <ul style="list-style-type: none"> - Saper analizzare i concetti dell'informatica dal punto di vista dell'apprendimento; 	<p>Part II - Nature and teaching of programming</p> <ul style="list-style-type: none"> - Introductory programming in the CC2001 models. - Imperative-first, functional-first and object-first approaches. - Programming paradigms from an educational perspective. - Environments and tools for learning programming. - Research on the teaching and learning of programming. <p>Dublin Descriptors</p> <p>Discipline-related skills</p> <p>1.1. Knowledge and understanding</p> <ul style="list-style-type: none"> - Knowledge of the historical and epistemological roots of computer science; - Knowledge of the main curricular models for teaching computing; - Knowledge of tools to support the learning of computing and programming; - Knowledge of assessment techniques to monitor the learning of the discipline; - Knowledge of the main research topics in computing and programming education. <p>1.2 Applying knowledge and understanding</p> <ul style="list-style-type: none"> - Being able to analyze the computing concepts from an educational viewpoint; - Being able to design simple learning units to introduce computing and programming concepts. <p>Soft skills</p>	
--	--	--	---	--

		<p>- Saper progettare semplici unità didattiche per introdurre concetti dell'informatica e della programmazione.</p> <p>Capacità trasversali Lo/la studente/ssa dovrà:</p> <p>2.1 Autonomia di giudizio</p> <p>- Saper analizzare criticamente i contenuti della disciplina dal punto di vista didattico;</p> <p>- Saper individuare un ordine adatto per affrontare gli argomenti previsti da una unità didattica;</p> <p>- Saper scegliere strumenti appropriati per favorire l'apprendimento di concetti o lo sviluppo di abilità nell'ambito dell'informatica e della programmazione.</p> <p>2.2 Abilità comunicative.</p> <p>- Essere in grado di presentare un concetto dell'informatica in forma comprensibile a un giovane discente.</p> <p>2.3 Capacità di apprendimento</p> <p>- Essere in grado di orientarsi nella letteratura sulla ricerca didattica in informatica;</p> <p>- Essere in grado di comprendere gli articoli di ricerca nella didattica dell'informatica.</p>	<p>2.1 Making judgements</p> <p>- Being able to analyze the topics of the discipline from an educational viewpoint;</p> <p>- Being able to schedule the content of a learning unit in a suitable order.</p> <p>- Being able to choose appropriate tools to support the learning of concepts or the development of skills in computing and programming.</p> <p>2.2 Communication skills.</p> <p>- Being able to present a computing concept in a suitable form for learning purposes.</p> <p>2.3 Learning skills</p> <p>- Being able to plan literature on computing education research;</p> <p>- Being able to understand research papers on computing education.</p>	
<p>Droni e sistemi robotici autonomi Drones and Autonomous Robotic Systems</p>	<p>ING- INF/05</p>	<p>Obiettivo del corso è fornire le basi teoriche e pratiche necessarie a comprendere e approcciare in modo pratico le recenti tecnologie dei sistemi robotici mobili, sensorizzati e dotati di autonomia totale o parziale, con particolare attenzione a droni aerei e agenti su ruota. Gli aspetti teorici e metodologici del corso riguarderanno la descrizione delle diverse classi di agenti robotici, i relativi aspetti funzionali e costruttivi, elementi</p>	<p>Specific educational objectives. The aim of the course is to provide the theoretical and practical bases for understanding and practically approaching the recent technologies related to mobile robotic systems. The course will focus on sensorized robotic agents, provided with total or partial autonomy, with particular attention to aerial drones and wheeled devices.</p>	

	<p>di dinamica e controllo delle unità mobili, la configurazione e il controllo di cluster di unità robotiche e swarming drones, le tipologie di sensori e le tecniche di elaborazione di segnali audio e video per l'analisi della scena acustico-visiva degli agenti robotici. Gli aspetti pratici del corso riguarderanno i diversi campi di applicazione delle tecnologie trattate e i principali ambienti software e linguaggi di programmazione per l'implementazione pratica di scenari applicativi. Saranno presi in esame alcuni casi di studio reali, impiegando dispositivi fisici disponibili nei laboratori didattici del corso. Lo/la studente/essa, alla fine del corso, dovrà:</p> <p>Capacità relative alle discipline:</p> <p>Conoscenza e comprensione</p> <ul style="list-style-type: none"> - conoscere i concetti e gli algoritmi fondamentali alla base del funzionamento e del controllo di semplici sistemi robotici mobili. - conoscere i concetti e gli algoritmi fondamentali alla base dell'analisi della scena mediante dati raccolti dai sensori e del design del comportamento autonomo. <p>Capacità di applicare conoscenza e comprensione</p> <ul style="list-style-type: none"> - saper analizzare e configurare i tipici scenari applicativi riguardanti agenti robotici mobili, operanti individualmente o in cluster coordinati. -saper utilizzare gli ambienti software e i linguaggio di programmazione utilizzati nel corso (ROS, Python). <p>Capacità trasversali /soft skills</p> <p>Autonomia di giudizio</p> <ul style="list-style-type: none"> -Saper valutare le caratteristiche di modelli e strumenti per la gestione di unità robotiche, e 	<p>The theoretical and methodological aspects of the course will concern the description of the different classes of robotic agents, the related functional and constructive aspects, elements of dynamics and control of mobile units, the configuration and control of clusters of robotic units and swarming drones, the types of sensors and audio and video signal processing techniques for the analysis of the visual and acoustic scene by the robotic agent. The practical aspects of the course will concern the different fields of application of the technologies covered and the main software environments and programming languages for the practical implementation of application scenarios. Some real case studies will be examined, using physical devices available in the didactic laboratories.</p> <p>At the end of the course, the student will have to:</p> <p>Applying knowledge and understanding</p> <ul style="list-style-type: none"> - know the basic concepts and algorithms underlying the operation and control of simple mobile robotic systems. - know the fundamental concepts and algorithms underlying the scene analysis using data collected by sensors and the design of autonomous behavior. <p>Capacità di applicare conoscenza e comprensione</p> <ul style="list-style-type: none"> - know how to analyze and configure the typical application scenarios concerning 	
--	---	--	--

		<p>saper effettuare una scelta personale del modello e degli strumenti software più adatti per la soluzione di un problema dato.</p> <p>-Saper distinguere tra diverse soluzioni per l'utilizzo di unità robotiche in una data circostanza e valutarne l'efficacia.</p> <p>Abilità comunicative</p> <p>-saper illustrare con rigore logico e terminologico, a voce e per iscritto, questioni tecniche inerenti la progettazione di applicazioni facenti uso di unità robotiche mobili e con diversi gradi di autonomia</p> <p>Capacità di apprendimento</p> <p>-saper reperire e utilizzare strumenti bibliografici e informatici utili per l'approfondimento autonomo di problemi inerenti la progettazione di applicazioni facenti uso di unità robotiche mobili e con diversi gradi di autonomia</p>	<p>mobile robotic agents, operating individually or in coordinated clusters.</p> <p>- know how to use the software environments and programming languages used in the course (ROS, Python).</p> <p>Soft skills</p> <p>Making judgements</p> <p>- know how to evaluate the characteristics of models and tools for the management of robotic units, and know how to make a personal choice of the model and the most suitable software tools for solving a given problem.</p> <p>- Know how to distinguish between different solutions for the use of robotic units in a given circumstance and evaluate their effectiveness.</p> <p>Communication skills</p> <p>- know how to illustrate with technical and terminological rigor, verbally and in writing, technical questions concerning the design of applications using mobile robotic units and with different degrees of autonomy</p> <p>Ability to autonomously</p> <p>- know how to find and use bibliographic and IT tools useful for the autonomous study of problems concerning the design of applications using mobile robotic units and with different degrees of autonomy</p>	
Foundations of neural networks	INF/01	<p>Scopo del corso</p> <p>Il corso sarà una introduzione alle Reti Neurali, con un focus specifico sulla loro storia, i loro successi/sconfitte ed il loro</p>	<p>Course aim</p> <p>The course will be an introduction to the Neural Networks, with a specific focus on their history, their successes / defeats and</p>	

	<p>status attuale da un punto di vista sia teorico che applicativo. Nella parte iniziale si osserveranno brevemente le reti neurali biologiche; si passerà quindi ad illustrare quali sono le componenti di base di una rete neurale e come queste vengono integrate nel modello. Successivamente passeremo a studiare il processo di learning nelle sue principali varianti (unsupervised/supervised, reinforced, etc.) e le più popolari strategie/topologie di reti. A seguire vedremo una galleria di modelli di rete (Jordan, Elman, Hopfield, Radial-Basis, etc.) e ne discuteremo i contesti applicativi per i quali risultano più efficaci</p> <p>1 Capacità relative alle discipline L'obiettivo principale è quello di introdurre i concetti di base, i risultati, la terminologia e, soprattutto, gli strumenti matematici comunemente utilizzati nel campo delle reti neurali oggi. Il campo è in forte e veloce espansione e richiede una ferma comprensione degli aspetti fondamentali e dei limiti computazionali della disciplina. In questo senso è essenziale una conoscenza di base sicura degli aspetti fondamentali dello strumento formale che le reti neurali rappresentano.</p> <p>1.1 Conoscenza e capacità di comprensione: L'obiettivo è la costruzione di un sistema integrato di conoscenze teoriche e pratiche, che consenta allo studente un'analisi</p>	<p>their current status from both a theoretical and application point of view. In the initial part, biological neural networks will be briefly observed; we will then go on to illustrate what are the basic components of a neural network and how they are integrated into the model. We will then move on to studying the learning process in its main variants (unsupervised / supervised, reinforced, etc.) and the most popular network strategies / topologies. Following we will see a gallery of network models (Jordan, Elman, Hopfield, Radial-Basis, etc.) and we will discuss the application contexts for which they are most effective.</p> <p>1 Sector-specific skills The main objective is to introduce the basic concepts, results, terminology and, above all, the mathematical tools commonly used in the field of neural networks today. The field is rapidly expanding and requires a firm understanding of the fundamental aspects and computational limits of the discipline. In this sense, a secure basic knowledge of the fundamental aspects of the formal tool that neural networks represent, is essential.</p> <p>1.1 Knowledge and comprehension skills:</p>	
--	---	---	--

		<p>critica del potenziale e dei limiti delle reti neurali.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: La capacità di analisi del potenziale delle reti neurali si dovrà coordinare con una comprensione dei principali contesti applicativi per le quali i diversi tipi di rete vengono oggi utilizzati nelle applicazioni.</p> <p>2 Capacità trasversali Molti problemi algoritmici vengono oggi affrontati e risolti utilizzando reti neurali. Cercheremo di illustrare e chiarire quali problemi si prestano ad una soluzione classica, da un punto di vista algoritmico, e quali invece è naturale vengano affrontati mediante l'uso di reti neurali.</p> <p>2.1 Autonomia di giudizio: L'obiettivo è quello di garantire la capacità di classificazione dei problemi che è opportuno affrontare mediante l'uso di reti neurali, di scelta di quale sia la rete più opportuna per un dato problema e di proposta di quali siano le tecniche più opportune per il suo training.</p> <p>2.2 Abilità comunicative: La terminologia nel campo delle reti neurali è un elemento essenziale per una efficace comunicazione. Questo sia dal punto</p>	<p>The goal is the construction of an integrated system of theoretical and practical knowledge, which allows the student a critical analysis of the potential as well as of the limits of neural networks.</p> <p>1.2 Knowledge and comprehension applied skills: The ability to analyze the potential of neural networks will be coordinated with an understanding of the main application contexts for which different types of networks are used today in applications.</p> <p>2 Cross-sectoral skills/soft skills Many algorithmic problems are being addressed and solved today using neural networks. We will try to illustrate and clarify which problems lend themselves to a classical solution, from an algorithmic point of view, and which instead is natural to address through the use of neural networks.</p> <p>2.1 Making judgements: The aim is to guarantee the ability to classify problems that should be addressed through the use of neural networks, to choose which is the most appropriate network for a given problem, and to propose which are the most appropriate techniques for his training.</p>	
--	--	---	--	--

		<p>di vista di una analisi ed illustrazione delle potenzialità, che da un punto di vista dei contesti applicativi per le quali si intende utilizzarle. In questo senso, obiettivo del corso sarà la cura di una attenta analisi dei termini da utilizzare nel comunicare gli aspetti relativi alle caratteristiche essenziali delle reti e dei problemi per le cui soluzioni queste vengono proposte.</p> <p>2.3 Capacità di apprendimento Si intende fornire agli allievi una adeguata serie di strumenti ed esempi atti a garantire loro la possibilità di studiare nuovi e originali modelli di rete e/o nuovi e originali modelli di training e di valutazione delle performances delle reti.</p>	<p>2.2 Communication skills: Terminology in the field of neural networks is an essential element for effective communication. This is the case both from the point of view of an analysis and illustration of the potential, and from the point of view of the application contexts for which it is intended to use them. In this sense, the objective of the course will be the care of a careful analysis of the terms to be used in communicating the aspects relating to the essential characteristics of the networks and the problems for which solutions are proposed.</p> <p>2.3 Learning skills: The aim is to provide students with an adequate set of tools and examples to guarantee them the possibility of studying new and original network models and / or new and original training models and evaluation techniques of network performances.</p>	
Geometria Computazionale	INF/01	<p>Il corso esplora, anche attraverso esempi e modelli semplificati, strutture di dati e tecniche algoritmiche di base per affrontare alcuni problemi significativi di geometria piana. I principali approcci introdotti sviluppano tecniche di tipo divide-et-impera, plane-sweep e incrementale-randomizzato.</p>	<p>The course explores, also by means of examples and simplified models, basic data structures and algorithmic techniques to approach noteworthy problems of planar geometry.</p>	

	<p>Particolare attenzione è rivolta all'analisi della correttezza e della complessità computazionale degli algoritmi discussi.</p> <p>Al termine del corso lo studente avrà acquisito la capacità di individuare tecniche appropriate per affrontare problemi nell'ambito della geometria computazionale e di valutarne criticamente potenzialità, efficacia, prestazioni e robustezza.</p> <p>Programma Sintetico</p> <p>Problemi di robustezza dell'elaborazione di dati geometrici.</p> <p>Approcci generali alla soluzione di problemi di geometria piana:</p> <ul style="list-style-type: none"> - Approccio divide-et-impera; - Tecniche di plane sweep; - Tecniche incrementali randomizzate. <p>Problemi notevoli nell'ambito della geometria computazionale piana e analisi dei costi computazionali:</p> <ul style="list-style-type: none"> - Convex hull; - Intersezioni di segmenti; - Partizioni di regioni poligonali: triangolazioni, mappe trapezoidali e suddivisioni monotone; - Problemi di point-location; - Problemi di prossimità: diagrammi di Voronoi e triangolazioni di Delaunay. <p>Capacità relative alle discipline</p> <p>Lo/la studente/essa dovrà:</p> <p>Capacità relative alla disciplina</p> <p>1.1. <i>Conoscenza e capacità di comprensione</i></p> <ul style="list-style-type: none"> - Conoscere alcuni approcci algoritmici di base della geometria computazionale e le relative applicazioni; 	<p>The main approaches being introduced exploit divide-et-impera, plane-sweep and randomized-incremental techniques.</p> <p>Particular attention is given to the analysis of the algorithms' correctness and computational costs.</p> <p>At the end of the course, the student will acquire the ability to identify appropriate techniques to address computational geometry problems and to assess their potential, effectiveness, performances and robustness.</p> <p>Short Syllabus</p> <p>Robustness issues of geometric data processing.</p> <p>General approaches to planar geometry problems:</p> <ul style="list-style-type: none"> - Divide-et-impera approach; - Plane-sweep techniques; - Randomized-incremental techniques. <p>Noteworthy problems of planar computational geometry and analysis of the related computational costs:</p> <ul style="list-style-type: none"> - Convex hull; - Segment intersection; - Partitions of polygonal regions: triangulations, trapezoidal maps and monotone subdivisions; - Point-location problems; - Proximity problems: Voronoi diagrams and Delaunay triangulations. <p>Sector-specific skills</p> <p>1.1. <i>Knowledge and understanding</i></p>	
--	---	---	--

		<ul style="list-style-type: none"> - Conoscere i costi computazionali di algoritmi che risolvono problemi notevoli di geometria piana; - Essere consapevole delle problematiche di robustezza dell'elaborazione di dati geometrici. <p><i>1.2 Capacità di applicare conoscenza e comprensione</i></p> <ul style="list-style-type: none"> - Saper individuare tecniche appropriate per affrontare problemi nuovi con gli strumenti della geometria computazionale; - Essere in grado di valutare i costi computazionali delle soluzioni progettate; - Essere in grado di sviluppare programmi di media dimensione per affrontare problemi di geometria piana. <p>Capacità trasversali / soft skills</p> <p><i>2.1 Autonomia di giudizio</i></p> <ul style="list-style-type: none"> - Saper analizzare i problemi al fine di identificare gli aspetti che si prestano ad essere affrontati con gli strumenti della geometria computazionale; - Essere in grado di confrontare e di valutare le implicazioni dell'applicazione di tecniche diverse alla risoluzione algoritmica di un problema geometrico. <p><i>2.2 Abilità comunicative.</i></p> <ul style="list-style-type: none"> - Essere in grado di usare un linguaggio preciso per presentare i risultati dell'analisi delle proprietà di un algoritmo. <p><i>2.3 Capacità di apprendimento</i></p> <ul style="list-style-type: none"> - Essere in grado di orientarsi nell'ambito della geometria computazionale e di studiare autonomamente nuovi algoritmi. 	<ul style="list-style-type: none"> - Knowledge of some basic computational geometry approaches and of their applications; - Knowledge of the computational costs of the algorithmic solutions of noteworthy planar problems; - Being aware of the robustness issues of geometric data processing. <p><i>1.2 Applying knowledge and understanding</i></p> <ul style="list-style-type: none"> - Being able to identify appropriate techniques to address new problems with the tools of computational geometry; - Being able to assess the computational costs of the designed solutions; - Being able to develop medium-scale programs to address planar problems. <p>Cross-sectoral skills/soft skills</p> <p><i>2.1 Making judgements</i></p> <ul style="list-style-type: none"> - Being able to analyze problems in order to identify what can be achieved with the tools of computational geometry; - Being able to compare and to assess the implications of different algorithmic techniques to solve a geometric problem. <p><i>2.2 Communication skills.</i></p> <ul style="list-style-type: none"> - Being able to use an accurate language to present the results of the analysis of an algorithm's properties. <p><i>2.3 Learning skills</i></p> <ul style="list-style-type: none"> - Being able to study autonomously new computational geometry algorithms. 	
Immagini e multimedialità	ING-INF/05	Obiettivi formativi specifici <i>Immagini e loro rappresentazioni</i>	Teaching goals <i>Images and their representations</i>	

	<p>Immagini statiche 2D: rappresentazioni numeriche, statistiche, grafiche. L'ambiente di elaborazione software MATLAB (©The Mathworks). <i>Processi di elaborazione delle immagini</i> Operatori matematici di tipo puntuale: trasformazioni. Operatori matematici in un intorno: filtraggi. Algoritmi per il miglioramento di qualità (enhancement) Segmentazione in regioni e stima di caratteristiche (feature) <i>Modellazione grafica 3D</i> L'ambiente di modellazione JAVA3D Trasformazioni geometriche e proiezioni Modellazione geometrica di oggetti Modelli di illuminazione e Interazione; animazione. Lo/la studente/essa dovrà: Capacità relative alle discipline 1.1 <i>Conoscenza e capacità di comprensione.</i> Conoscere i formalismi matematici per la rappresentazione delle immagini statiche: nozioni di algebra dei vettori e delle matrici; elementi di statistica; nozioni di geometria discreta. Conoscenza di elementi di programmazione vettoriale. Conoscenza di elementi dell'ambiente di elaborazione numerica e grafica MATLAB. Tecniche base di filtraggio di immagini per miglioramento di qualità ed estrazione di caratteristiche. Modellazione a oggetti in ambito di grafica computerizzata. 1.2 <i>Capacità di applicare conoscenza e comprensione.</i> Saper lavorare in un ambiente di</p>	<p>Static 2D images: numerical, statistical, graphical representations. The MATLAB (©The Mathworks) software environment. <i>Image Processing Algorithms</i> Point operators: image transformations. Neighborhood operators: image digital filtering. Image enhancement techniques Region segmentation and feature extraction. <i>Graphic 3D Modelling</i> The JAVA3D™ modelling environment Geometric transforms Geometric object modelling Lighting and shadowing Interactions; animations. The student is expected to: Sector-specific skills 1.1 <i>Knowledge and understanding.</i> Be acquainted with the mathematical formalisms representing still 2D images: vector and matrix algebra; basic statistics; basic discrete geometry. Basics of vector programming. Be familiar with the MATLAB programming environment. Be acquainted with image filtering and feature extraction techniques in the space domain. Be familiar with the JAVA3D modelling environment and geometric transforms. 1.2 <i>Applying knowledge and understanding.</i> Be able to work into an image processing software environment. Be able to select and use numerical-</p>	
--	--	--	--

		<p>elaborazione di immagini digitali. Saper utilizzare librerie numeriche e statistiche per l'analisi delle immagini in presenza di rumore. Saper usare tecniche di filtraggio digitale e valutare la sensibilità al rumore degli algoritmi. Saper progettare un modello grafico 3d a partire dalla modellazione concettuale (scenegraph). Saper realizzare il modello tramite un linguaggio a oggetti.</p> <p>Capacità trasversali / soft skills</p> <p>2.1 <i>Autonomia di giudizio.</i> Saper analizzare un modello di dominio di interesse per il WS. Saper individuare adeguati formalismi di modellazione e linguaggi di implementazione per lo sviluppo di un'applicazione WS. Saper selezionare l'ambiente software e le librerie per lo sviluppo di un progetto.</p> <p>2.2 <i>Abilità comunicative.</i> Essere in grado di motivare per iscritto le scelte di modellazione concettuale e implementative. Capacità di apprendimento: Saper approfondire in modo autonomo le caratteristiche di un problema di analisi delle immagini e modellazione 3D, nell'ambito delle tematiche studiate a lezione. Saper affrontare lo studio di aggiornamenti in ambito di standard tecnologici e ambienti di sviluppo.</p>	<p>statistical libraries for real image processing. Be able to use digital filtering algorithms, as well as evaluate their performances in terms of robustness to noise and efficiency. Be able to design a graphical 3D model adopting an object-oriented approach. Be able to implement the model with object-oriented libraries.</p> <p>Cross-sectoral/soft skills</p> <p>2.1 <i>Making judgements.</i> Be able to critically analyze an image processing technique. Be able to select the appropriate software tools and libraries.</p> <p>2.2 <i>Communication skills.</i> Be able to express with a technical report the design and implementation choices of an image processing, graphical modelling problem.</p> <p>2.3 <i>Learning skills.</i> Be able to master newer developments in terms of technological standards and software development environments for image processing and model building.</p>	
Informatica e diritto	INF/01	<p>Il corso di Informatica e diritto si propone di introdurre gli studenti ai problemi specifici legati all'applicazione delle metodologie e tecnologie informatiche in ambito giuridico. Gli studenti sono portati a comprendere argomenti basilari di tipo legale ed etico relativi alla professione di informatico, ivi compreso il loro ruolo individuale in tali ambiti.</p>	<p>The Computer Science and Law course aims to introduce students to the specific problems related to the application of the methodologies and information technologies in the legal field. Students are led to understand basic legal and ethical topics related to the profession of</p>	

	<p>Argomenti principali:</p> <ul style="list-style-type: none"> ● elementi di base di diritto ● e-government, codice dell'amministrazione digitale, strategia digitale italiana ed europea ● e-legislation, informatica applicata alla formazione e gestione dei testi legislativi ● e-democracy, open-government, e-vote ● diritti in internet (privacy, anonimato, diritto d'autore) ● digital forensic e reati informatici ● aspetti etici dell'informatica <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione Conoscere le caratteristiche di base del dominio giuridico, sia dal punto di vista della rappresentazione ed utilizzo della conoscenza giuridica, che dell'organizzazione dei servizi della Pubblica Amministrazione e della loro informatizzazione.</p> <p>Conoscere funzioni e criticità dei sistemi informatici applicati alla Pubblica Amministrazione, alle istituzioni democratiche e all'attività forense. Comprendere gli aspetti etici legati alla digitalizzazione.</p> <p>1.2 Capacità di applicare conoscenza e comprensione Saper valutare e progettare applicazioni informatiche nei campi dell'e-government, e-legislation ed e-democracy.</p> <p>Saper identificare ed utilizzare gli standard e le tecnologie applicabili in un determinato contesto giuridico.</p> <p>Capacità trasversali / soft skills Lo/la studente/ssa dovrà:</p>	<p>computer science, including their individual role in these areas.</p> <p>main topics:</p> <ul style="list-style-type: none"> ● basic elements of law ● e-government, code of digital administration, Italian and European digital strategy ● e-legislation, information technology applied to training and management of legislative texts ● e-democracy, open-government, e-vote ● internet rights (privacy, anonymity, copyright) ● digital forensic and computer crimes ● ethical aspects of computer science <p>At the end of the course the student is expected to:</p> <p>Sector-specific skills</p> <p>1.1 Knowledge and understanding: To know the basic aspects of the legal domain, both from the point of view of the representation and use of legal knowledge, and of the organization of the Public Administration services and their computerisation.</p> <p>Knowing the functions and critical aspects of IT systems applied to the Public Administration, democratic institutions and legal activities. Understanding the ethical aspects related to digitization.</p> <p>1.2 Applying knowledge and understanding:</p>	
--	--	---	--

		<p>2.1 Autonomia di giudizio Acquisire la capacità di valutazione rispetto alle tecnologie e agli approcci di innovazione applicabili in relazione al contesto giuridico.</p> <p>2.2 Abilità comunicative Conoscere i termini giuridici in modo da essere in grado di interagire correttamente e comprenderne le esigenze alla base dei requisiti di un sistema di informatico applicato in contesto giuridico.</p> <p>2.3 Capacità di apprendimento Imparare ad essere autonomo nell'espandere le proprie conoscenze di informatica giuridica oltre le nozioni e gli esempi appresi ed analizzati durante il corso. Essere anche in grado di informarsi autonomamente consultando la letteratura scientifica.</p>	<p>To be able to evaluate and design IT applications in the fields of e-government, e-legislation and e-democracy. To be able to identify and use standards and technologies suitable for a given legal context.</p> <p>Cross-sectoral skills/soft skills At the end of the course the student is expected to:</p> <p>2.1 Making judgements: Acquire the capacity to assess technologies and innovation approaches appropriate to the legal environment.</p> <p>2.2 Communication skills: Know the legal terms to be able to interact properly and understand the needs underlying the requirements of a computer system applied in a legal context.</p> <p>2.3 Learning skills: Learn to be autonomous in expanding his/her knowledge of juridical informatics beyond the notions and examples learned and analyzed during the course. Also, be able to inform himself/herself by consulting scientific literature.</p>	
Information retrieval	ING-INF/05	<p>L'information Retrieval (IR) è una disciplina che è importante storicamente e che ha ricevuto un forte impulso in seguito all'avvento del Web. Il corso mira a presentare gli aspetti concettualmente più importanti dei sistemi d'IR, con particolare attenzione ai motori di ricerca sul Web. Il corso è diviso nelle due parti seguenti:</p> <p>1. IR classico:</p>	<p>Information Retrieval (IR) is a discipline that has a high historical importance and has received an even increased attention after the coming of the Web. The course aims to present the main conceptual issues underlying IR systems, with particular emphasis on Web search engines. The course is divided into the two following parts;</p>	

		<ul style="list-style-type: none"> ● le interfacce utente per l'IR (classificazione, rassegna); ● i modelli formali dell'IR (Booleano, spazio vettoriale, probabilistico e varianti quali BM25, Language models); ● la struttura di un indice invertito (aspetti di base, compressione); ● la classificazione (definizione, classificatori naive di Bayes); ● il clustering (algoritmi gerarchici e approssimati); ● la valutazione (fondamenti, metodologie, metriche; aspetti di ricerca). <p>2. Web IR:</p> <ul style="list-style-type: none"> ● il grafo del Web (dimensioni e forma: reti piccolo mondo, a invariata di scala, forma a papillon); ● l'analisi dei link per il ranking a altre applicazioni (PageRank, HITS, varianti); ● il crawling (concetti e architettura); ● spam (cenni); ● architettura di un motore di ricerca (cenni). <p>Vengono inoltre presentati casi di studio e approfondimenti.</p> <p>Capacità relative alle discipline</p> <p>1.1 Conoscenza <i>e capacità di comprensione</i>: lo studente acquisisce le conoscenze di base sull'Information retrieval e sulle problematiche connesse. Egli inoltre conosce sia gli argomenti</p>	<p>1. Classical IR:</p> <ul style="list-style-type: none"> ● user interfaces for IR (classification, survey); ● formal IR models (Boolean, vector space, probabilistic and variants as BM25, Language models); ● structure of the inverted index (basics, compression); ● classification (definition, naive Bayes classifiers) ● clustering (hierarchical and approximate algorithms); ● evaluation (foundations, methodologies, metrics; research topics). <p>2. Web IR:</p> <ul style="list-style-type: none"> ● Web graph (size and shape: small world and scale-free networks, bow-tie shape); ● link analysis for ranking and other applications (PageRank, HITS, variants); ● crawling (concepts and architecture); ● spam (short account); ● search engine architecture (short account). <p>Furthermore, some case studies and specific issues are presented and discussed.</p> <p>Sector-specific skills</p> <p>1.1 Knowledge <i>and understanding</i>: the student will acquire the basic knowledge of the IR field and of related topics. The</p>	
--	--	---	--	--

		<p>di base sia le correnti linee di ricerca e le tendenze future della disciplina.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: lo studente impara ad analizzare, comprendere e valutare i modelli principali dei sistemi d'IR, per quanto concerne sia l'IR classico sia l'IR sul Web. Acquisisce inoltre conoscenze sulle tecniche di valutazione dei sistemi d'IR e impara a scegliere fra le varie metodologie di valutazione.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: lo studente acquisisce una capacità di valutazione critica sulle tecnologie, gli algoritmi e le tecniche che possono influire positivamente o negativamente sull'uso corretto ed efficace dei sistemi d'IR.</p> <p>2.2 Abilità comunicative: lo studente impara a descrivere in modo tecnicamente corretto ed usando la terminologia appropriata lo stato dell'arte dell'IR. Sa inoltre evidenziare anche le connessioni con altri corsi riguardanti le tecnologie web per il cloud, la scienza dei dati, l'internet of things, il machine learning e il social computing.</p> <p>2.3 Capacità di apprendimento: lo studente impara ad essere autonomo nell'espandere le proprie conoscenze sull'IR oltre le nozioni e gli esempi appresi ed analizzati durante il corso. È anche in grado di informarsi autonomamente consultando la letteratura scientifica.</p>	<p>student will know both basic topics and advanced research trends of the field, with some hints at future research developments.</p> <p>1.2 applying knowledge and understanding: the student learns to analyze, understand, and evaluate the main IR models, for both classic and Web IR. The student will also know the evaluation techniques of IR system effectiveness, as well as learn how to choose among the various evaluation methodologies.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements: the student acquires the skills to critically evaluate how technologies, algorithms, and techniques can positively or negatively affect a correct and effective use of IR systems.</p> <p>2.2 Communication skills: the student learns how to describe in a technically appropriate way, and using the correct terminology, the state of the art of the IR discipline. The student also can emphasize the relationships with other courses, concerning web/cloud technologies, data science, internet of things, machine learning, and social computing.</p> <p>2.3 Learning skills: the student learns how to autonomously expand his/her own knowledge of IR concepts beyond the notions and examples that have been presented during the lectures. The student</p>	
--	--	---	--	--

			is also able to autonomously obtain information from the scientific literature.	
Information visualization	INF/01	<p>Obiettivo principale del corso è far conoscere allo studente le diverse tecniche di rappresentazione e di presentazione delle informazioni a partire dai dati grezzi. Le conoscenze acquisite verranno utilizzate per sviluppare abilità e competenze nella creazione pratica di visualizzazioni efficaci di dati.</p> <p>Capacità relative alle discipline - Conoscenza e comprensione Lo/la studente/essa acquisisce specifiche conoscenze dei principali concetti e principi teorici relativi alla visualizzazione delle informazioni a partire dai dati grezzi. Lo/la studente/essa impara inoltre a scegliere fra diverse tecniche di rappresentazione e presentazione delle informazioni, insieme alla conoscenza delle potenzialità aggiuntive fornite da un approccio interattivo. - Capacità di applicare conoscenza e comprensione Lo/la studente/essa acquisisce specifiche capacità di applicare in maniera critica la comprensione e le conoscenze maturate sulla visualizzazione delle informazioni. Lo studente sarà in grado di applicare la comprensione e le conoscenze acquisite alle diverse tipologie di dati e ai diversi domini informativi.</p> <p>Capacità trasversali /soft skills - Autonomia di giudizio Lo/la studente/essa acquisisce la capacità di valutare criticamente le caratteristiche che</p>	<p>The main objective of the course is to introduce the student to the different techniques of representation and presentation of information, starting from the raw data. The acquired knowledge will be used to develop skills and competences in the practical creation of effective data visualizations.</p> <p>Sector-specific skills - Knowledge and understanding The student acquires specific knowledge of the main concepts and theoretical principles related to the visualization of information from raw data. The student also learns to choose between different techniques of information representation and presentation, together with knowledge of the additional potential provided by an interactive approach. - Applying knowledge and understanding The student acquires specific skills to critically apply the understanding and knowledge gained on information visualization. The student will be able to apply the acquired understanding and knowledge to the different types of data and information domains.</p> <p>Cross-sectoral skills/soft skills - Making judgements The student acquires the ability to critically evaluate the characteristics that</p>	

		<p>possono influire positivamente o negativamente sull'utente per ciò che concerne la visualizzazione delle informazioni e l'interazione con esse. Acquisirà inoltre la capacità di giudizio necessaria per valutare l'appropriatezza delle diverse tecniche apprese nell'insegnamento in diversi contesti applicativi.</p> <ul style="list-style-type: none"> - Abilità comunicative <p>Lo/la studente/essa acquisisce la conoscenza dei fattori principali che rendono efficace la comunicazione attraverso visualizzazione delle informazioni e potrà quindi riutilizzare tali conoscenze anche nell'ambito della comunicazione interpersonale e professionale.</p> <ul style="list-style-type: none"> - Capacità di apprendimento <p>Lo/la studente/essa acquisisce inoltre specifiche capacità trasversali relative all'abilità di reperire ed utilizzare risorse informatiche, scientifiche e bibliografiche per l'approfondimento autonomo delle tematiche introdotte durante il corso.</p>	<p>can positively or negatively affect the user with regard to the display of information and interaction with it. He or she will also acquire the necessary judgement to assess the appropriateness of the different techniques learned for different application contexts.</p> <ul style="list-style-type: none"> - Communication skills <p>The student acquires knowledge of the main factors that make communication effective through the visualization of information and can therefore reuse this knowledge also in interpersonal and professional communication.</p> <ul style="list-style-type: none"> - Learning skills <p>The student also acquires specific soft skills related to the ability to find and use computer, scientific and bibliographic resources for the autonomous study of the topics introduced during the course.</p>	
Ingegneria del software - Progettazione e laboratorio	ING-INF/05	<p>Obiettivo del corso è l'approfondimento delle principali tematiche dell'Ingegneria del Software. Viene richiamato il ciclo di progettazione e di vita di un prodotto software. Vengono poi studiati e approfonditi gli argomenti più avanzati, quali l'ingegneria del software esistente, il configuration management, le metodologie di test, la progettazione dei sistemi, le metriche teoriche ed operative, i più noti modelli di qualità di prodotto e di processo, le metodiche di pianificazione e controllo, i principali strumenti di lavoro.</p> <p>Capacità relative alle discipline</p>	<p>The target of the course is to examine the principal elements of Software Engineering. We consider the design and life cycle of a software product studying the more advanced topics: configuration management, existing software engineering, test methodologies, theoretic and practical metrics, most famous product and process quality models, planning and control methodologies, principal working tools.</p> <p>Knowledge and Understanding: The course aims to provide knowledge to organize and manage the software development processes.</p>	

	<p>Conoscenza e comprensione: l'insegnamento si propone di fornire le conoscenze per organizzare, trattare e gestire processi di sviluppo del software</p> <p>Al termine del corso lo studente dovrà essere in grado di:</p> <ul style="list-style-type: none"> • gestire l'organizzazione di un processo ingegnerizzato di sviluppo del software e di un gruppo di lavoro; • utilizzare tecniche avanzate di test • utilizzare le principali metriche operative del software • utilizzare strumenti di valutazione della qualità di processo e di prodotto nell'ambito del software • conoscere le caratteristiche le caratteristiche principali degli strumenti CASE di sviluppo del software. <p>Capacità trasversali:</p> <ul style="list-style-type: none"> • Autonomia di giudizio: lo studente dovrà essere in grado di valutare autonomamente quali siano le caratteristiche degli strumenti e delle metodologie applicate nei diversi contesti gestione e organizzazione di un progetto di sviluppo e manutenzione del software. • Abilità comunicative: lo studente dovrà acquisire la capacità e le tecniche di interazione in un gruppo di lavoro per lo sviluppo del software. • Capacità di apprendimento: lo studente dovrà essere in grado di apprendere in maniera autonoma tecniche di processo, di misurazione e di qualità alternative a quelle descritte nel corso 	<p>Skills related to the disciplines: After completing the course the student will be able to:</p> <ul style="list-style-type: none"> • manage the organization of a software development engineered process and of working group • use advanced testing techniques • use the principal software metrics • use assessment tools of software process and product quality models • know CASE tools main features. <p>Soft skills:</p> <ul style="list-style-type: none"> • Judgment autonomy: students must be able to assess which are the characteristics of the tools and methodologies applied in different management and organization contexts to develop and maintain a software project • Communication skills: the student must acquire the techniques of interaction in a software development working group • Learning skills: the student will be able to learn autonomously technology, measurement and quality processes alternative to those described in the course 	
--	---	---	--

Intelligenza artificiale	ING- INF/05	<p>Gli obiettivi del corso sono principalmente introduttivi, concettuali e culturali:</p> <ul style="list-style-type: none"> - Conoscere i concetti e i principi fondamentali dell'Intelligenza Artificiale (IA) e di alcuni dei suoi maggiori settori, quali i sistemi basati sulla conoscenza, le tecniche di rappresentazione della conoscenza e di ragionamento, l'elaborazione del linguaggio naturale, il problem solving e l'apprendimento automatico - Capire che l'IA è una disciplina ingegneristica, volta a sviluppare sistemi software in grado di svolgere compiti cognitivi avanzati. - Capire relazioni e differenze tra gli approcci tradizionali dell'informatica e l'approccio IA. - Conoscere le principali caratteristiche dei due approcci all'IA: simbolico e sub-simbolico - Conoscere alcune delle principali aree applicative dell'IA <p>Capacità relative alle discipline Lo studente dovrà:</p> <ol style="list-style-type: none"> 1. <i>Conoscenza e capacità di comprensione:</i> acquisire specifiche conoscenze dei principali concetti e principi dell'IA. Conoscere e sapere utilizzare le tecniche base per la rappresentazione della conoscenza e dell'apprendimento automatico, anche mediante specifiche esercitazioni laboratoriali 	<p>The course objectives are mainly introductory, conceptual, and cultural:</p> <ul style="list-style-type: none"> - Introducing basic concepts and principles of Artificial Intelligence (AI) and of some of its major areas, such as knowledge-based systems, knowledge representation and reasoning, natural language processing, problem solving, and machine learning; - Understanding how AI is to be considered an engineering discipline, aimed at developing software systems capable of performing advanced cognitive tasks - Understanding relationships and differences between the Traditional approach of Computer science and the Artificial Intelligence Approach - Knowing the major characteristics of the symbolic approach to AI and of the sub-symbolic approach to AI - Knowing some of the major application areas of AI <p>Sector-specific skills The student will have to:</p> <ol style="list-style-type: none"> 1. <i>Knowledge and understanding:</i> acquiring specific knowledge of the main concepts and basic principles of AI. Knowing and exploiting basic techniques for knowledge 	
--------------------------	----------------	---	--	--

		<p>2. <i>Capacità di applicare conoscenza e comprensione</i>: sapere come analizzare e rappresentare conoscenza specifica di un dominio, come rappresentare il significato di semplici testi in linguaggio naturale e saper risolvere semplici problemi di machine learning e saper applicare le conoscenza di cui sopra in specifici contesti applicativi.</p> <p>Capacità trasversali / soft skills Lo studente dovrà:</p> <p>1. <i>Autonomia di giudizio</i>: saper valutare indipendentemente le caratteristiche di un'applicazione informatica ed essere in grado di capire se si tratta di un dominio adatto all'IA o alle tecniche tradizionali.</p> <p>2. <i>Abilità comunicative</i>: saper illustrare con rigore logico e terminologico, a voce e per iscritto, lo scenario generale e le modalità di funzionamento di un sistema software basato su tecniche di IA.</p> <p>3. <i>Capacità di apprendimento</i>: essere in grado di apprendere le nozioni base dell'IA, al fine poi eventualmente di raffinare e di approfondire specifici settori della disciplina.</p>	<p>representation and machine learning, also by means of some specific laboratory class. Knowing what is a conceptual model of a specific problem-solving task.</p> <p>2. <i>Applying knowledge and understanding</i>: knowing how to analyze and represent domain specific knowledge, how to represent the meaning of a simple Natural Language text and how to approach simple machine learning projects, and how to apply the above-mentioned knowledge in specific application context.</p> <p>Cross-sectoral skills/soft skills The student will have to:</p> <p>1. <i>Making judgments</i>: being able to independently evaluate the characteristics of a computer application and to be able to understand if the domain is adequate for the traditional approach or for the AI approach.</p> <p>2. <i>Communication skills</i>: acquiring the ability to describe effectively and through appropriate models the scenario of an AI based system and its advantages over a traditional system</p> <p>3. <i>Learning skills</i>: being able to learn the basics of AI, in order to possibly later refine and deepen specific areas of the discipline.</p>	
Interactive 3D Graphics	ING-INF/05	Dai dispositivi mobili fino ai sistemi per la Realtà Virtuale, sempre più applicazioni si basano sulla	From mobile devices to virtual reality systems, more and more applications are	

	<p>capacità di visualizzare, in modo realistico, oggetti e scene tridimensionali con cui gli utenti possono interagire: videogiochi, esperienze immersive, visualizzazioni di prodotti e di progetti sono gli esempi più noti.</p> <p>Il corso introduce i principali concetti, algoritmi e tecnologie nel campo della grafica 3D interattiva, con esempi pratici in WebGL (tramite la libreria three.js) e Unity. In particolare, partendo dalla pipeline per il rendering interattivo, si esamina in dettaglio il suo funzionamento: definizione di geometrie, trasformazioni, generazione di frammenti, e loro assemblaggio in nell'immagine finale. Si passa poi alla simulazione dei materiali e degli effetti dell'illuminazione, esaminando nel dettaglio le equazioni per il <i>Physically-Based Rendering</i> (e la loro implementazione tramite linguaggi di shading) oggi ampiamente utilizzate nei videogiochi, nella produzione cinematografica e nella realtà virtuale. Infine, si trattano aspetti strettamente correlati al rendering, come le tecniche di animazione e le strutture dati spaziali.</p> <p>Argomenti principali:</p> <ul style="list-style-type: none"> • Il ciclo per il rendering 3D interattivo. La pipeline per il rendering in tempo reale. • Rappresentazione di geometrie. • Trasformazioni affini. Proiezioni ortografiche e prospettiche. • Rasterizzazione e interpolazione. Aliasing e metodi di anti-aliasing. • Shader programmabili. Il linguaggio glsl. • Physically-Based Shading. L'equazione generale di rendering. BRDF lambertiana e micro-facet. 	<p>based on the capability to visualize 3D objects and scenes, faithfully simulate their aspect, and allow users to interact with them. Notable examples include videogames, immersive experiences, product and design visualization.</p> <p>The course introduces the main concepts, algorithms and technologies in the field of interactive 3D graphics, with practical examples in WebGL (through the three.js library) and Unity. More specifically, starting from the interactive 3D rendering pipeline, we examine in detail its functioning: geometry representation, transformations, rasterization, and fragments merging into the final image. Then, we focus on how to simulate the effect of lighting on materials, presenting the equations for <i>Physically-Based Rendering</i> (and their implementation through shading languages) that are nowadays popular in videogames, movie production, and virtual reality. Finally, we also cover topics that are strictly related to rendering, such as animation techniques and spatial data structures.</p> <p>Main topics:</p> <ul style="list-style-type: none"> • The interactive 3D rendering cycle. The real-time rendering pipeline. • Geometry representation. • Affine transformations. Perspective and orthographic projections. • Rasterization and interpolation. Aliasing and anti-aliasing methods. 	
--	---	--	--

	<ul style="list-style-type: none"> • Tecniche di shading: material mapping, bump mapping, reflection mapping, refraction mapping, environment mapping, shadow mapping. • Rendering basato su immagini. Effetti di post-processing. • Animazioni tramite keyframing, skeleton-based, basate su leggi fisiche. Sistemi di particelle. <p>Capacità relative alle discipline Lo/la studente/studentessa dovrà:</p> <p>1.1. Conoscenza e capacità di comprensione Durante il corso, lo studente acquisisce le conoscenze necessarie per comprendere il funzionamento di un'applicazione basata su grafica 3D interattiva (videogiochi, applicazioni di realtà virtuale, visualizzazioni 3D). Inoltre, sa valutarne le prestazioni ed individuare come migliorarle.</p> <p>1.2 Capacità di applicare conoscenza e comprensione Grazie ai numerosi esempi ed esercizi, e alle attività di progettazione e implementazione richieste per l'esame, lo studente acquisisce la capacità di progettare e implementare un'applicazione basata su grafica 3D interattiva, anche Web-based, scegliendo le tecnologie e gli algoritmi più adatti al caso specifico.</p> <p>Capacità trasversali / soft skills</p> <p>2.1 Autonomia di giudizio Lo studente acquisisce una capacità di valutazione critica sulle tecnologie, gli algoritmi e le tecniche di programmazione che possono influire positivamente o negativamente sulla</p>	<ul style="list-style-type: none"> • Programmable shaders. The glsl language. • Physically-Based Shading. General rendering equation. Lambertian and micro-facet BRDF. • Shading techniques: material mapping, bump mapping, reflection mapping, refraction mapping, environment mapping, shadow mapping. • Image-based rendering. Post-processing effects. • Animation techniques: keyframing, skeleton-based, physics-based. Particle systems. <p>Sector-specific skills</p> <p>1.1. Knowledge and understanding During the course, the student learns how to understand the functioning of an application based on interactive 3D graphics (videogames, virtual reality applications, 3D visualizations). Moreover, he/she knows how to evaluate and improve the rendering performances.</p> <p>1.2 Applying knowledge and understanding Thanks to several examples and exercises, and to the projects that are due for the exam, the student learns how to design and implement an application based on interactive 3D graphics, Web-based or not, choosing the algorithms and technologies that are more suited to the case at hand.</p> <p>Cross-sectoral/soft skills</p>	
--	--	--	--

		<p>corretta ed efficace implementazione di un'applicazione basata su grafica 3D.</p> <p>2.2 Abilità comunicative. Lo studente impara a descrivere in modo tecnicamente corretto, ed usando la terminologia appropriata, un'applicazione o una tecnica della grafica 3D interattiva.</p> <p>2.3 Capacità di apprendimento Lo studente impara ad essere autonomo nell'espandere le proprie conoscenze oltre le nozioni e gli esempi appresi ed analizzati durante il corso, acquisendo le conoscenze di base per accedere alla letteratura tecnico / scientifica relativa ad argomenti avanzati.</p>	<p>2.1 Making judgements The student learns how to critically evaluate the technologies, algorithms and programming techniques that can determine the correct and effective implementation of an application based on interactive 3D graphics.</p> <p>2.2 Communication skills The student learns how to describe, in technically suitable terms, an application or a technique in the field of interactive 3d graphics.</p> <p>2.3 Learning skills The student learns how to become autonomous in expanding his/her knowledge beyond the concepts and examples that are given in class, by acquiring the basic knowledge which is necessary to access the technical and scientific literature about advanced topics.</p>	
Linguaggi e compilatori	INF/01	<p>Questo corso, intende fornire una conoscenza delle caratteristiche dei vari paradigmi di programmazione, concentrandosi sui principi "universali" che guidano la progettazione, realizzazione e implementazione dei moderni linguaggi di programmazione, ponendo particolare enfasi nello studio delle relazioni tra tecniche di progetto, semantica ed implementazione dei linguaggi di programmazione.</p> <p>Questo corso complementa i corsi base sui linguaggi di programmazione delle lauree triennali in informatica relativi ai paradigmi imperativo e funzionale, introducendo la teoria e la pratica di programmazione nei paradigmi</p>	<p>This course aims to provide a knowledge of the features of the various programming paradigms, focusing on the "universal" principles that guide the design, development and implementation of modern programming languages, with special emphasis on the study of relations between project techniques, semantics and implementation of programming languages.</p> <p>This course complements the basic courses on programming languages of the three-year degrees in computer science related to imperative and functional paradigms, introducing the theory and</p>	

	<p>logico e logico-funzionale, dove caratteristiche quali high-order e non-determinismo possono essere sfruttate per scrivere codice compatto, elegante e facilmente riutilizzabile. Inoltre si analizzano i principi operazionali che stanno alla base dei vari linguaggi dichiarativi e come questi guidino nella implementazione delle relative macchine astratte.</p> <p>Per completare il quadro della implementazione dei linguaggi, vengono affrontate le principali problematiche, soluzioni e tecniche concernenti il front-end di un compilatore: si presentano gli algoritmi per l'analisi lessicale e sintattica che sono alla base di tool come Lex e Bison (per il linguaggio C) o Alex e Happy (per il linguaggio Haskell). Inoltre si presentano i principali metodi per l'analisi di semantica statica dei programmi, mediante grammatiche di attributi e sistemi di tipi. Infine viene illustrata la generazione del codice intermedio (senza ottimizzazioni).</p> <p>Il corso ha come scopo ultimo quello di sviluppare uno spirito critico che permetta di arrivare ad una programmazione consapevole in cui saper scegliere il paradigma più adatto a seconda del problema applicativo da risolvere, sapendo quali costrutti di un linguaggio usare, e a quale costo complessivo (in termini di risorse impiegate in tutta la pipeline di implementazione).</p> <p>Capacità relative alle discipline</p> <p><i>1.1 Conoscenza e capacità di comprensione:</i> Lo studente conosce i principali concetti della programmazione nei principali paradigmi dei linguaggi di programmazione. Conosce inoltre i principi operazionali che stanno alla base delle</p>	<p>practice of programming in logic and functional-logic paradigms, where features such as high-order and non-determinism can be exploited to write compact, elegant and easily reusable code. It also analyzes the operational principles underlying the various declarative languages and how these guide the implementation of their corresponding abstract machines.</p> <p>To complete the picture of the implementation of languages, the main issues, solutions and techniques regarding the front-end of a compiler are addressed: the algorithms for lexical and syntactic analysis, that are the base of tools such as Lex and Bison (for the language C) or Alex and Happy (for the Haskell language), are shown. In addition, the main methods for program's static semantics analysis are illustrated, by means of attributed grammars and type systems. Finally, intermediate code generation (without optimizations) is presented.</p> <p>The course has as its ultimate aim to develop a critical spirit that allows to arrive at a conscious programming in which one knows how to choose the most appropriate paradigm depending on the application problem to be solved, knowing what constructs of a language to use, and at what total cost (in terms of resources used throughout the implementation pipeline).</p> <p>Sector-specific skills</p> <p><i>1.1 Knowledge and understanding</i></p>	
--	---	--	--

		<p>implementazioni dei vari linguaggi. Conosce infine le principali problematiche, soluzioni e tecniche concernenti il front-end di un compilatore.</p> <p><i>1.2 Conoscenza e capacità di comprensione applicate:</i></p> <p>Lo studente impara nuovi linguaggi di programmazione dichiarativi e impara a scrivere codice nei vari paradigmi usando lo stile e le tecniche appropriate. Inoltre impara a conoscere ed usare degli strumenti che tipicamente vengono usati per scrivere un compilatore ma hanno ricadute applicative in moltissimi altri ambiti informatici.</p> <p>L'attività di progetto permette allo studente di consolidare le conoscenze teoriche presentate durante le lezioni frontali attraverso il loro utilizzo in casi reali, promettendogli quindi di sviluppare autonomamente un front-end di un semplice linguaggio imperativo.</p> <p>Capacità trasversali/soft skills</p> <p><i>2.1 Autonomia di giudizio:</i></p> <p>Obiettivo del corso è principalmente quello di sviluppare uno spirito critico che permetta di saper scegliere il linguaggio di programmazione più adatto a seconda del problema applicativo da risolvere, sapendo quali costrutti di un linguaggio usare, e a quale costo. Inoltre si intende migliorare la capacità dello studente di identificare le soluzioni più pratiche e semplici nell'implementazione di sistemi software.</p> <p><i>2.2 Abilità comunicative:</i></p> <p>Lo studente impara l'esatto significato dei termini usati nella programmazione e nella implementazione dei linguaggi.</p>	<p>The student knows the main concepts of programming in the main paradigms of programming languages. He also knows the operational principles underlying the implementations of the various languages. Finally, he knows the main problems, solutions, and techniques relating to the front-end of a compiler.</p> <p><i>1.2Applying knowledge and understanding</i></p> <p>Students learn new declarative languages and learn to write code in the various paradigms by using appropriate styles and techniques. They also learn to know and to use the tools that are typically used to write a compiler but can have also many applications in other information technology areas.</p> <p>The project activity allows students to consolidate the theoretical knowledge presented during lectures through its use in real cases, thus allowing them to independently develop a front-end of a simple imperative language.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>2.1Making judgements</i></p> <p>The course objective is primarily to develop a critical spirit that allows one to choose the most appropriate programming language depending on the application problem to be solved, knowing which constructs of a language to use, and at what cost. It also aims to improve the student's ability to identify the most practical and simple solutions in the implementation of software systems.</p>	
--	--	---	---	--

		<p>Attraverso l'attività di progetto di gruppo lo studente migliora le proprie capacità comunicative e di interazione.</p> <p><i>2.3 Capacità di apprendere:</i></p> <p>Le conoscenze apprese permettono allo studente muoversi con una certa disinvoltura tra la miriade di linguaggi di programmazione riuscendo a inquadrare velocemente le caratteristiche di nuovo linguaggio, e a impararlo in breve tempo.</p> <p>Grazie all'interazione con i compagni del gruppo di progetto lo studente impara a valutare il proprio grado di apprendimento mediante il confronto con gli altri.</p>	<p><i>2.2 Communication skills</i></p> <p>The student learns the exact meaning of the terms used in programming language coding and implementation.</p> <p>Through the group project activity, students improve their communicative and interaction skills.</p> <p><i>2.3 Learning skills</i></p> <p>The knowledge acquired allows students to move with ease among the myriad of programming languages, and to learn new languages in a short time by quickly framing new language features.</p> <p>Thanks to the interaction with the other members of the project team, students learn to evaluate their degree of learning by comparing with others.</p>	
<p>Logica per l'informatica Logic for applications</p>	<p>INF/01</p>	<p>Obiettivi Formativi Specifici.</p> <p>Il corso vuole fornire un'introduzione alla teoria dei modelli finiti, un ambiente formale adeguato per la fondazione di concetti che vengono da varie parti dell'informatica, come le basi di dati, la complessità computazionale e la teoria dei linguaggi.</p> <p>Obiettivo del corso è quello di evidenziare come le applicazioni informatiche sopra menzionate abbiano spinto alla creazione di tecniche e risultati propri della teoria dei modelli finiti, che non hanno un corrispettivo nella teoria dei modelli classica.</p> <p>Ad esempio, mostreremo come le logiche su modelli finiti siano il punto di partenza per sviluppare linguaggi di interrogazione di base di dati e come la teoria dei modelli finiti sia utilizzata per</p>	<p>Specific educational objectives</p> <p>This course aims to provide an introduction to the theory of finite models, a formal environment for the foundation of concepts coming from various parts of Computer Science, such as database theory, computational complexity and formal languages.</p> <p>Objective of the course is to highlight how the Computer Science applications mentioned above account for the developments of techniques and results which are specific to final model theory, and do not have a correspondence in the classical model theory. For example, we will show how logics over finite models are the starting point for developing database query</p>	

		<p>provare risultati sulla loro espressività e complessità.</p> <p>Capacità relative alle discipline</p> <p>1.1 Conoscenza e capacità di comprensione Conoscere le tecniche ed i risultati propri della teoria dei modelli finiti.</p> <p>1.2 Conoscenza e capacità di comprensione applicate Saper riconoscere la logica più adatta alla formalizzazione di una specifica interrogazione, ed essere in grado di formalizzare correttamente tale proprietà. Saper applicare le tecniche di espressività studiate per stabilire se una data interrogazione è esprimibile in una data logica.</p> <p>Capacità trasversali</p> <p>2.1 Autonomia di giudizio Saper riconoscere una formalizzazione corretta di un dato problema. Saper valutare quale strumento utilizzare per formalizzare un specifico problema.</p> <p>2.2 Abilità comunicative. Saper comunicare le proprie intuizioni e dimostrazioni ai compagni di corso. Riuscire a formalizzare un problema nel modo più semplice possibile e riuscire a spiegare tale formalizzazione in modo comprensibile. Scrivere in modo formalmente corretto la soluzione degli esercizi.</p>	<p>languages, and finite model theory techniques are used for proving results about their expressiveness and complexity.</p> <p>Framework skills</p> <p>1.1 Knowledge and understanding To know techniques and results specific to the theory of finite models.</p> <p>1.2 Applying knowledge and understanding To be able to recognize the logic which is more suited for formalizing a specific query and to perform such a formalization correctly. To be able to apply the studied techniques in order to prove that a specific query is formalizable or not in a specific logic.</p> <p>Transversal Skills</p> <p>2.1 Making judgements To be able to recognize a correct formalization. To be able to evaluate which tool is more suited to solve a specific problem.</p> <p>2.2 Communication skills To know how to communicate the intuition behind a formalization of a problem to class mates. To formalize a problem using the simplest possible solution and to be able to explain this solution informally. To be able to write a formally correct solution of an exercise.</p>	
--	--	---	--	--

		<p>2.3 Capacità di apprendimento Essere in grado di approfondire in maniera autonoma la conoscenza del ruolo della teoria dei modelli finiti in Informatica.</p>	<p>2.3 Learning skills To be able to further develop and deepen, by independent study, the knowledge of the role of finite</p>	
<p>Metodi formali per l'informatica Formal methods</p>	<p>INF/01</p>	<p>L'obiettivo del corso di Metodi Formali è quello di fornire allo studente strumenti rigorosi per la verifica formale di correttezza di programmi. In particolare, verrà illustrata la Logica di Hoare, basata su invarianti, per la verifica di programmi sequenziali, paralleli e distribuiti. Si discuteranno formalmente correttezza e terminazione di diversi algoritmi sequenziali classici. Mentre questioni come assenza di deadlock e fairness verranno affrontate formalmente per una serie di problemi paradigmatici di concorrenza, quali produttore-consumatore, trasmissione di dati.</p> <p>Capacità relative alla disciplina</p> <p>1.1 Conoscenza e capacità di comprensione: conoscere e comprendere le principali tecniche usate per descrivere formalmente la semantica di programmi e per verificarne la correttezza, sia nel caso di programmi sequenziali che nel caso di programmi concorrenti (paralleli e distribuiti).</p> <p>1.2 Conoscenza e capacità di comprensione applicate: saper applicare tecniche per la verifica formale di proprietà di programmi.</p> <p>Capacità trasversali/soft skills</p>	<p>The aim of the course on Formal Methods is to illustrate rigorous tools for the formal verification of program correctness. In particular, the Hoare Logic, based on</p> <p>The aim of the course on Formal Methods is to illustrate rigorous tools for the formal verification of program correctness. In particular, the Hoare Logic, based on</p> <p>The aim of the course on Formal Methods is to illustrate rigorous tools for the formal verification of program correctness. In particular, the Hoare Logic, based on</p> <p>invariants, will be presented for the verification of sequential, parallel and distributed programs. Correctness and termination of a number of classical sequential algorithms will be discussed formally, while issues such as deadlock and fairness will be formally analyzed for a number of paradigmatic problems of concurrency, such as Producer-Consumer, Data Transmission.</p> <p>Sector-specific skills</p>	<p>Programmazione e laboratorio Logica Matematica Sistemi Operativi. Programming Logic Operating Systems</p>

		<p>2.1 Autonomia di giudizio: lo studente impara a essere cosciente delle ambiguità presenti nelle descrizioni informali, e a usare tecniche di formalizzazione per la verifica di correttezza.</p> <p>2.2 Abilità comunicative: lo studente impara a formalizzare e a descrivere in maniera rigorosa il comportamento atteso di programmi sequenziali e concorrenti.</p> <p>2.3 Capacità di apprendimento: il corso fornisce le nozioni fondamentali dei metodi formali per la verifica di correttezza dei programmi.</p>	<p>1.1 Knowledge and understanding: to know and understand the principal techniques used to formally describe the semantics of programs, and to study the correctness of sequential and concurrent programs, both parallel and distributed.</p> <p>1.2 Applying knowledge and understanding: to know how to apply techniques for the formal verification of properties of programs.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1. Making judgments: the student learns how to recognise ambiguities in informal descriptions, and how to use formal techniques to verify program correctness.</p> <p>2.2 Communication skills: the students learns how to formalise and describe in a rigorous way the expected behaviour of sequential and concurrent programs.</p> <p>2.3 Learning skills: fundamental notions concerning formal methods are given for studying and verifying program correctness.</p>	
<p>Progettazione di applicazioni mobili Mobile application design</p>	INF/01	<p>Il corso vuole fornire le competenze necessarie alla progettazione ed alla realizzazione di applicazioni e servizi per dispositivi mobili quali smartphone e tablet, con particolare riferimento al design dell'interazione. Verranno esplorati gli aspetti peculiari che caratterizzano i dispositivi mobili, verranno analizzate le principali</p>	<p>The course aims to provide the necessary skills to design and develop applications and services for mobile devices such as smartphones and tablets, with a specific focus on interaction design. We will explore the unique features of mobile devices, we will analyze the most</p>	

	<p>piattaforme mobili esistenti, verranno studiate tecniche per la progettazione dell'interazione con applicazioni e servizi all'utente su dispositivi mobili, verrà esplorato lo sviluppo di applicazioni mobili per le principali piattaforme mobili.</p> <p>Capacità relative alle discipline</p> <ul style="list-style-type: none"> - Conoscenza e comprensione <p>Lo/la studente/essa acquisisce durante l'insegnamento di Progettazione di Applicazioni Mobili: specifiche conoscenze delle principali tecnologie utilizzabili per la realizzazione di applicazioni/servizi per dispositivi mobili; comprensione delle fasi fondamentali della progettazione dell'interazione con applicazioni mobili, in particolar modo per quanto riguarda l'approccio basato sul ciclo envision-prototype-evaluate; conoscenza dei pattern d'interfaccia più comunemente utilizzati per l'interazione con applicazioni/servizi per dispositivi mobili; conoscenza degli aspetti implementativi fondamentali per lo sviluppo di applicazioni/servizi mobili.</p> <ul style="list-style-type: none"> - Capacità di applicare conoscenza e comprensione <p>Lo/la studente/essa acquisisce durante l'insegnamento di Progettazione di Applicazioni Mobili: specifiche capacità di identificare quali tecnologie siano più appropriate allo sviluppo di applicazioni/servizi mobili nel contesto di mercato considerato; capacità di progettare l'interazione con applicazioni/servizi mobili sfruttando l'approccio leggero basato sul ciclo envision-prototype-evaluate; capacità di saper scegliere i pattern d'interfaccia più appropriati</p>	<p>important mobile platforms, we will study techniques used to design the interaction with applications and services for mobile devices, we will explore the development of mobile applications for the main mobile platforms.</p> <p>Sector-specific skills</p> <ul style="list-style-type: none"> - Knowledge and understanding <p>During the Mobile Application Design course, students will acquire: specific knowledge of the available technologies for the development of applications and services for mobile devices; knowledge of the key phases in the design of interaction with mobile applications, with special reference to the envision-prototype-evaluate cycle; knowledge of the most common interface pattern for interaction with applications and services for mobile devices; knowledge of the core implementation details for the development of mobile applications and services.</p> <ul style="list-style-type: none"> - Applying knowledge and understanding <p>During the Mobile Application Design course, students will acquire: the ability to identify the most appropriate technologies for the development of applications and services for mobile devices in the considered market scenario; the ability to design the interaction with applications and services for mobile devices using the envision-prototype-evaluate cycle; the ability to choose the most appropriate</p>	
--	---	---	--

		<p>alla specifica applicazione/servizio mobile considerati, evitando l'utilizzo di comuni anti-pattern; capacità di sviluppare applicazioni/servizi mobili interattivi per le principali piattaforme mobili.</p> <p>Capacità trasversali/soft skills - Autonomia di giudizio Lo/la studente/essa acquisirà con l'insegnamento di Progettazione di Applicazioni Mobili la capacità di valutare autonomamente e criticamente l'appropriatezza di specifiche soluzioni d'interfaccia per l'interazione con applicazioni e servizi per dispositivi mobili. Acquisirà inoltre la capacità di giudizio necessaria per valutare l'appropriatezza delle diverse tecniche di progettazione e implementative apprese nell'insegnamento a diversi tipi di applicazioni/servizi mobili. - Abilità comunicative Lo/la studente/essa acquisirà durante l'insegnamento di Progettazione di Applicazioni Mobili l'abilità di comunicare, tramite progettazione grafica su carta, la propria visione dell'interazione utente con applicazioni/servizi mobili, oltre a descrivere tale visione mediante redazione di una relazione tecnica su assignment e attività di progetto assegnati durante il corso. - Capacità di apprendimento Lo/la studente/essa acquisirà inoltre specifiche capacità di identificare problematiche di interazione sul campo con interfacce mobili mediante valutazione utente delle soluzioni proposte durante gli assignment ed il progetto.</p>	<p>interface patterns for the considered mobile application/service, avoiding common antipatterns; the ability to develop interactive mobile applications/services.</p> <p>Cross-sectoral skills/soft skills - Making judgements By attending the Mobile Application Design course, students will acquire the ability to autonomously and critically evaluate the appropriateness of specific interface solutions for the interaction with mobile applications and services. Moreover, they will acquire the ability to evaluate the appropriateness of the specific design and development techniques learned in the course to different mobile applications/services. - Communication skills During the Mobile Application Design course, students will acquire the ability to communicate their vision of interaction with mobile applications/services through graphical design on paper, in addition to being able to present such vision through technical reports on the course assignments and final project. - Learning skills Students will also acquire the ability to identify interaction issues with mobile interfaces through user evaluation in the field of the solutions proposed during the assignments and final project.</p>	
--	--	--	---	--

<p>Progettazione di sistemi multimodali e laboratorio Multimodal Systems Design and Laboratory</p>	<p>ING- INF/05</p>	<p>Il corso ha quattro obiettivi fondamentali: i) approfondire l'aspetto strutturale e funzionale dei sistemi multimodali sia dal punto di vista teorico sia operativo; ii) fornire criteri e metodi per affrontare il problema della selezione e della integrazione di materiali multimodali; iii) presentare alcune metodologie recenti per la progettazione di sistemi multimodali orientate alla esperienza dell'utente (Design for Experience) e all'adattamento culturale (Cross-Cultural Design); iv) fornire concetti e metodi per la valutazione della qualità e degli effetti sociali e culturali dei sistemi multimodali. L'approccio seguito integra concetti derivanti dagli studi sulla composizione di materiali computazionali, dalla semiotica sociale e dalle teorie narrative per studiare come creare strutture di significati attraverso la combinazione nello spazio e nel tempo di testi, immagini, oggetti sonori, video e audiovisivi. L'attività di laboratorio è finalizzata alla analisi e alla valutazione di casi di studio e alla applicazione dei concetti e dei modelli teorici nella realizzazione (individuale o in gruppo) di prodotti multimodali concreti. Alla fine del corso lo/la studente/essa dovrà:</p> <p>Capacità relative alla disciplina Conoscenza e capacità di comprensione Conoscere le concettualizzazioni di base e i vocabolari tecnici per l'analisi, la progettazione e la valutazione di prodotti multimodali. Conoscere i principali approcci, metodi e modelli per affrontare un problema di progettazione multimodale.</p>	<p>The course has four main objectives: i) to study the structural and functional features of multimodal artifacts from a theoretical and operative point of view; ii) to provide the students with criteria and methods for addressing the problem of selection and integration of multimodal materials; iii) to present and critically discuss some recent approaches for multimodal design proposed within the field of "Design for Experience" and "Cross-cultural Design"; iv) to discuss various conceptualizations and methods for the assessment of the quality and the social and cultural implications of multimodal systems. The approach is inspired to recent results obtained within the fields of material studies (e.g., computational materials, materiality, material compositions), social semiotics and narrative theories and focuses on meaning construction and interpretation through the space-time combination of texts, images, sound objects, videos and audio-visuals. Laboratory activity is aimed at the analysis and evaluation of case studies and the application of theoretical knowledge to concrete design problems of multimodal products.</p> <p>At the end of the course, the student should be able:</p> <p>Sector-specific skills Knowledge and understanding</p>	
--	------------------------	---	---	--

		<p>Conoscenze e capacità di comprensione applicate Saper applicare le conoscenze concettuali e metodologiche di cui sopra a situazioni di progettazione reali relative a diversi generi di prodotti multimodali, diverse tipologie di utenti e di contesti d'uso.</p> <p>Capacità trasversali/soft skills Autonomia di giudizio Saper analizzare, confrontare e valutare in maniera critica soluzioni di progetto da diversi punti di vista (strutturale, funzionale, comunicativo, sociale/culturale), secondo diversi criteri (fattibilità tecnica, usabilità, accessibilità, efficacia comunicativa, conseguenze etiche e sociali) e a diversi livelli di aggregazione (intero prodotto, parti del prodotto rispetto al tutto, prodotto rispetto al contesto d'uso) Abilità comunicative Saper spiegare come un dato prodotto multimodale soddisfa (o non soddisfa) gli scopi comunicativi per cui è stato progettato. Saper spiegare e giustificare le decisioni tecniche e di progetto. Saper ascoltare e capire i punti di vista degli altri: saper collaborare in un gruppo.</p> <p>Capacità di apprendimento Essere capaci di generalizzare (specializzare) concetti, modelli e metodi. Saper stabilire analogie tra concettualizzazioni diverse e saper trasferire concetti e metodi tra domini differenti. Saper utilizzare i meccanismi di base della inferenza (deduzione, abduzione e induzione) per sintetizzare soluzioni di progetto e verificarne la adeguatezza rispetto ai requisiti.</p>	<p>To know the conceptualizations and technical vocabularies for the analysis, design and evaluation of multimodal products. To know the main approaches, methods and models for addressing a design problem in the multimodal field.</p> <p>Applying knowledge and understanding To effectively exploit the aforementioned knowledge and methods in concrete design situations characterized by different multimodal genres, different typologies of audience and different use contexts.</p> <p>Cross-sector skills/ soft skills Making judgements To compare, evaluate and criticize two or more design solutions from different perspectives (e.g., structural, behavioural, functional, and socio-cultural points of view), according to different criteria (e.g., usability, accessibility, communicative effectiveness, socio-cultural effects) and at different aggregation and abstraction levels (e.g., the whole product, single parts with respect to the whole, the product with respect to the context of use).</p> <p>Communication skills To describe a design solution and to explain why and how it achieves the intended goals and requirements. To be able to motivate and support a point of view but also to listen to and understand</p>	
--	--	--	---	--

		Essere capaci di riflettere sul proprio percorso di progettazione (meta riflessione).	<p>the others: to be able to collaborate within design teams.</p> <p>Learning skills To generalize (or specialize) concepts, models and methods. To establish analogies between different conceptualizations, in order to transfer and apply design knowledge from an application domain to another. To be able to use the basic inference mechanisms (e.g., abduction, induction and deduction) for the synthesis and evaluation of design solutions. To be able to reflect on personal design processes (meta-reflection).</p>	
Progettazione e analisi orientate agli oggetti	ING-INF/05	<p>Il corso di Progettazione e Analisi Orientate agli Oggetti (PAOO) mira a presentare gli aspetti avanzati e concettualmente più importanti della realizzazione di sistemi software secondo il paradigma Object Oriented (OO). Viene presentata soprattutto l'attività di Progettazione ma, seppure con minore dettaglio, viene discussa anche la fase di Analisi. Verranno descritti: il linguaggio di modellazione UML (Unified Modeling Language), i principi di costruzione di un buon software OO, i design pattern, il refactoring. Verranno anche discussi alcuni casi di studio di problemi reali. Il programma dettagliato è il seguente:</p> <ul style="list-style-type: none"> ● UML per il progetto. I diagrammi UML usati per la progettazione: diagrammi di classe, di collaborazione, di sequenza, degli stati, di attività, dei componenti e del dispiegamento (deployment). 	<p>The course on Object Oriented Analysis and Design (OOAD) aims to present the advanced issues, as well as the most important topics from a conceptual viewpoint, of the implementation of software systems according to the Object Oriented (OO) paradigm. The focus is on the Design activity but, though with minor details, also the Analysis phase is discussed. Main topics are: UML (Unified Modeling Language), principles leading to good quality OO design, design patterns, and refactoring. In addition, case studies from real world problems will be discussed. The detailed syllabus is:</p> <ul style="list-style-type: none"> ● UML for design. UML diagrams used for software design: class, collaboration, sequence, state, activity, 	

		<ul style="list-style-type: none"> ● I principi della progettazione OO. I concetti e le tecniche che consentono di costruire progetti OO di buona qualità, ad esempio: domini, ingombro, coesione, spazio degli stati, comportamento, conformità di tipo, progetto per contratti, ecc. ● I design pattern. I design pattern stanno alla progettazione OO come la programmazione strutturata sta alla programmazione imperativa classica. I pattern sono schemi di soluzioni ricorrenti, la cui conoscenza semplifica il progetto di nuovo software OO. Vengono presentati e analizzati i pattern principali. ● UML per l'analisi. Viene completata la presentazione di UML, introducendo i diagrammi di classe concettuali e dei casi d'uso. Vengono discusse le principali tematiche dell'analisi OO con UML. ● Il Refactoring. Viene presentata l'attività del refactoring, volta a migliorare codice già esistente e funzionante. ● Casi di studio e approfondimenti. I concetti e le tecniche introdotti vengono esemplificati in situazioni reali, attraverso un'attività seminariale che può essere svolta, opzionalmente, anche dagli studenti. <p>Capacità relative alle discipline 1.1 Conoscenza e <i>capacità di comprensione</i>: lo studente acquisisce le conoscenze dei concetti e</p>	<p>component, and deployment diagrams.</p> <ul style="list-style-type: none"> ● OO Design principles. Concepts and techniques leading to good OO design, for instance: domains, encumbrance, cohesion, state space, behavior, type conformance, design by contract, etc. ● Design patterns. Design patterns are for OO design what structured programming is for classical imperative programming. Patterns are recurrent solution schema and their knowledge simplifies the design of new software. Main patterns are presented and discussed. ● UML for analysis. The UML description is concluded by presenting conceptual class and case use diagrams. The main issues of OO analysis with UML are discussed. ● Refactoring. The refactoring activity, aimed at improving already existing and working code, is described. ● Case studies. The concepts and techniques introduced during the course are exemplified in real world cases, by means of seminars (by invited experts or, optionally, by students). <p>Sector-specific skills</p>	
--	--	--	--	--

		<p>delle tecniche avanzati di analisi e progetto del mondo OO.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: lo studente saprà progettare software OO applicando i principi di buona progettazione e usando opportunamente i design pattern.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: lo studente acquisisce una capacità di valutazione critica sulle tecnologie, gli algoritmi e le tecniche che possono influire positivamente o negativamente sull'uso corretto ed efficace del paradigma OO, sia a livello di progettazione sia a livello di analisi. Sa inoltre giudicare il livello di qualità di differenti scelte progettuali.</p> <p>2.2 Abilità comunicative: lo studente impara a descrivere in modo tecnicamente corretto ed usando la terminologia appropriata le scelte progettuali di un software OO. Sa inoltre evidenziare pregi e difetti di progetti software, anche realizzati da altri.</p> <p>2.3 Capacità di apprendimento: lo studente impara ad essere autonomo nell'espandere le proprie conoscenze di PAOO. È anche in grado di comprendere un progetto software e di apprendere nuovi design pattern.</p>	<p>1.1 Knowledge and understanding: the student will know the advanced concepts and techniques for Object Oriented (OO) analysis and design.</p> <p>1.2 Applying knowledge and understanding: the student will be able to design OO software, applying OO good design principles and using design patterns appropriately.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements: the student acquires critical evaluation skills on technologies, algorithms, and techniques that can positively or negatively affect a correct and effective use of the OO paradigm, at both design and analysis levels. The student is also able to judge the quality of different design choices.</p> <p>2.2 Communication skills: the student learns to describe in a technically appropriate way, and using the correct terminology, the design choices of an OO software. The student also can highlight positive and negative aspects in software projects, even when implemented by third parties.</p> <p>2.3 Learning skills: the student learns to be autonomous in expanding his/her own knowledge of OOAD. The student is also able to understand a software project and to learn new design patterns.</p>	
<p>Programmazione su architetture parallele Programming on parallel architectures</p>	<p>INF/01</p>	<p>Questo corso fornisce una introduzione alla programmazione parallela per moderne architetture multi- e many-core,</p>	<p>This course provides an introduction to parallel programming for modern multi- and many-core architectures, illustrating</p>	

		<p>illustrando i paradigmi basati su message passing e su shared memory. Si forniranno le nozioni di base sui sistemi paralleli e distribuiti e sulle tecniche per la loro programmazione. In particolare verranno affrontati: programmazione parallela su GPU; programmazione in ambiente MPI; programmazione shared memory con OpenMP.</p> <p>Al fine di seguire le lezioni con profitto è richiesta la conoscenza della programmazione in linguaggio C e dei concetti base relativi ai sistemi operativi e alla architettura degli elaboratori.</p> <p>Capacità relative alla disciplina</p> <p>Conoscenza e capacità di comprensione: lo studente acquisisce conoscenze di base sulla programmazione parallela e distribuita, sulle problematiche inerenti alla programmazione su architetture parallele/distribuite, e sui principi di progettazione di applicazioni in ambienti con comunicazione basata su shared memory e/o message passing.</p> <p>Capacità di applicare conoscenza e comprensione: Anche attraverso esercizi pratici, lo studente impara a valutare ed a scegliere l'adozione del framework e delle scelte progettuali più adatte alla soluzione di un problema, anche in relazione all'hardware a disposizione, e a progettare una sua soluzione.</p>	<p>the paradigms based on message passing and shared memory. The basics on parallel and distributed systems and on the techniques for their programming will be considered. In particular, the following will be addressed: parallel programming on GPU; programming in MPI environment; shared memory programming with OpenMP.</p> <p>In order to follow lessons successfully, knowledge of programming in C language and of the basic concepts on operating systems and computer architecture is required.</p> <p>Sector-specific skills</p> <p>Knowledge and understanding: the student will acquire basic notions on the programming of parallel and distributed architectures, on the specific design difficulties involved by these architectures, and on the basic programming techniques adopted in shared-memory and message-passing frameworks.</p> <p>Applying knowledge and understanding: Also through practical exercises, the student learns to evaluate and choose the adoption of the framework and the design</p>	
--	--	---	---	--

		<p>Capacità trasversali/soft skills</p> <p>Autonomia di giudizio: lo studente acquisisce capacità di valutazione critica delle tecniche di programmazione e degli algoritmi impiegati nella soluzione di un problema in ambiente parallelo/distribuito.</p> <p>Abilità comunicative: lo studente impara a comunicare con terminologia appropriata, le principali nozioni della programmazione parallela/distribuita, nei contesti degli approcci basati su shared memory e message passing.</p> <p>Capacità di apprendimento: lo studente sarà in grado di mettere a frutto le conoscenze di base acquisite, per estendere autonomamente le proprie competenze relativamente a più complessi framework paralleli/distribuiti e moderne architetture hardware.</p>	<p>choices that are most suitable for solving a problem, also in relation to the hardware at hand, and to design its own solution.</p> <p>Cross-sectorial skills/soft skills</p> <p>Making judgements: the student acquires critical evaluation skills of the programming techniques and algorithms needed to solve a problem in a parallel/distributed environment.</p> <p>Communication skills: the student learns to communicate, with appropriate terminology, the main notions about parallel/distributed programming, in the contexts of approaches based on shared memory and message passing.</p> <p>Learning skills: the student will be able to exploit the basic knowledge he/she acquired, to independently extend his/her skills in relation to more complex parallel/distributed frameworks and modern hardware architectures.</p>	
Quantum computing and communication	INF/01	<p>Programma:</p> <p>1) Introduzione e panoramica Bit quantistici Algoritmi quantistici Elaborazione sperimentale dell'informazione quantistica Informazione quantistica</p>	<p>Syllabus:</p> <p>1) Introduction and overview Quantum bits Quantum algorithms Experimental quantum information processing Quantum information</p>	

		<p>2) Introduzione alla meccanica quantistica Algebra lineare I postulati della meccanica quantistica Applicazione: codifica superdensa L'operatore densita' Decomposizione e purificazioni di Schmidt EPR e disuguaglianze di Bell</p> <p>3) Circuiti quantistici Algoritmi quantistici Operazioni a singolo qubit Operazioni controllate Misurazione Porte quantistiche universali Riepilogo del modello di calcolo con circuiti quantistici</p> <p>4) La trasformata quantistica di Fourier e le sue applicazioni La trasformata quantistica di Fourier Stima della fase Applicazioni: ricerca di ordine e fattorizzazione Applicazioni generali della trasformata quantistica di Fourier</p> <p>5) Algoritmi di ricerca quantistica L'algoritmo di ricerca quantistica Ricerca quantistica come simulazione quantistica Conteggio quantistico Accelerare la soluzione dei problemi NP-completi Ricerca quantistica di un database non strutturato Ottimalita' dell'algoritmo di ricerca Limiti degli algoritmi black box</p> <p>6) Computer quantistici: realizzazione fisica</p> <p>- Conoscenza e capacita' di comprensione</p>	<p>2) Introduction to quantum mechanics Linear algebra The postulates of quantum mechanics Application: superdense coding The density operator The Schmidt decomposition and purifications EPR and the Bell inequalities</p> <p>3) Quantum circuits Quantum algorithms Single qubit operations Controlled operations Measurement Universal quantum gates Summary of the quantum circuit model of computation</p> <p>4) The quantum Fourier transform and its applications The quantum Fourier transform Phase estimation Applications: order-finding and factoring General applications of the quantum Fourier transform</p> <p>5) Quantum search algorithms The quantum search algorithm Quantum search as a quantum simulation Quantum counting Speeding up the solution of NP-complete problems Quantum search of an unstructured database Optimality of the search algorithm Black box algorithm limits</p> <p>6) Quantum computers: physical realization</p>	
--	--	---	---	--

		<p>Conoscere i principi base delle tecnologie basate sulla fisica quantistica, le principali possibilità e comprensione delle differenze rispetto ai modelli di calcolo non-quantistici.</p> <p>- Conoscenza e capacità di comprensione applicate</p> <p>Schematizzazione di circuiti quantistici e loro utilizzo in algoritmi semplici, ma rappresentativi delle potenzialità. Capacità di utilizzo delle principali porte quantistiche.</p> <p>- Autonomia di giudizio Saper giudicare autonomamente le potenzialità e i limiti della computazione quantistica allo stato attuale delle tecnologie.</p> <p>- Abilità comunicative Capacità di esporre e comunicare le particolarità del calcolo e della comunicazione quantistica in ambito non fisico in presentazioni o brevi tesine.</p> <p>- Capacità di apprendere Apprendere concetti di ambito fisico e metterli in relazione al linguaggio e agli strumenti informatici.</p>	<p>- Knowledge and understanding</p> <p>Knowing the basic principles of based technologies on quantum physics, the main possibilities and understanding the differences with respect to non-quantum calculation models</p> <p>- Applying knowledge and understanding</p> <p>Schematization of quantum circuits and their use in simple algorithms, but representative of the potential applications. Ability to use the main quantum gates.</p> <p>- Making judgements; Knowing how to judge the potential and the limits of quantum computing with current technologies independently.</p> <p>- Communication skills Ability to exhibit and communicate peculiarities of computing and quantum communication to a non-physicist audience in presentations or short term papers.</p> <p>- Ability to learn Learning concepts of quantum physics and ability to relate them computer science language and tools.</p>	
Ragionamento automatico	INF/01	Il corso affronta lo studio del ragionamento automatico nell'ambito dell'Intelligenza	Automated Reasoning will be introduced according to IEEE/ACM classification as a	

	<p>Artificiale. Aderenti alle sotto-aree IEEE/ACM di "Intelligent Systems" saranno introdotti i principali linguaggi per la rappresentazione della conoscenza e le principali tecniche per il ragionamento su di essa. In particolare sarà posta particolare enfasi allo studio dei linguaggi logici, quali ad esempio il cosiddetto Answer Set Programming, agli action description languages adatti a definire problemi di planning e al linguaggio Minizinc per la descrizione di problemi vincolati. Tali linguaggi vengono oggi utilizzati con successo nella risoluzione di problemi combinatorici, di ottimizzazione, e nella modellazione di sistemi intelligenti, con ricadute multidisciplinari. Saranno inoltre presentate e comparate diverse tecniche per la ricerca di soluzioni e di ottimi in spazi vincolati.</p> <p>I paradigmi saranno presentati in modo rigoroso sia per quanto riguarda gli aspetti semantici ed implementativi che sotto il profilo pratico, mediante la codifica e risoluzione di diversi problemi.</p> <p>Programma Constraint Satisfaction/Optimization Problems (CSP/COP) e principali tecniche matematiche ed informatiche per affrontarli. Constraint Programming: propagazione di vincoli, risolutori di vincoli, vincoli globali. Programmazione a vincoli pratica usando Minizinc. Programmazione logica, rappresentazione della conoscenza, e ragionamento automatico: Answer Set Programming (ASP). Modellazione di problem usando ASP.</p>	<p>subfield of "Intelligent Systems".</p> <p>The main languages and techniques for knowledge representation and reasoning will be presented, focusing in particular on logic languages such as Answer Set Programming, Action Description Languages for planning, and the constraint modeling language Minizinc. Those languages are commonly used for modeling and solving combinatorial optimization problems, and for programming intelligent systems in a multi-disciplinary context. The main techniques for constraint-based solution search will be presented and compared. Paradigms will be presented either at a theoretical level or at a practical level, modeling and solving several benchmark problems.</p> <p>Program Constraint Satisfaction/Optimization Problems (CSP/COP) and main techniques used for dealing with them. Constraint Programming: constraint propagation, constraint solver, global constraints. Practical Constraint programming with Minizinc. Logic programming and Knowledge Representation and Automated Reasoning: Answer Set Programming (ASP). Practical problem definition and solving with ASP.</p>	
--	---	---	--

	<p>Relazioni tra espressività e complessità di linguaggi per la rappresentazione della conoscenza. Action Description Languages e Planning.</p> <p>Capacità relative alle discipline</p> <p>1.1 <i>Conoscenza e capacità di comprensione:</i> lo studente approfondirà le proprie capacità di analizzare a risolvere problemi. In particolare, vedrà una panoramica dei linguaggi usati in Intelligenza Artificiale per modellare i problemi e le principali tecniche dell'Intelligenza Artificiale per trovare le soluzioni ai problemi codificati. Nel corso si enfatizzerà l'attenzione verso i linguaggi logici per la rappresentazione della conoscenza e per il ragionamento.</p> <p>1.2 <i>Conoscenza e capacità di comprensione applicate:</i> Lo studente sarà in grado di scegliere il linguaggio opportuno e di modellare in tale linguaggio problemi concreti quali ad esempio problemi combinatorici, di ottimizzazione, di allocazione risorse, di ragionamento automatico, che emergono quotidianamente nelle realtà industriali.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 <i>Autonomia di giudizio:</i> Dalle specifiche formali di un problema lo studente sarà in grado di capire se sia conveniente codificarlo nei linguaggi visti nel corso e, nel caso, di farlo. In particolare, se il problema risultasse NP completo, lo studente sarà in grado di codificarlo in modo compatto e di sfruttare le euristiche inserite nel corso degli anni nei risolutori dei linguaggi per una soluzione il più possibile</p>	<p>Relationships between expressivity and complexity. Action Description Languages and Planning.</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding:</i> the student will improve her/his capability of analyzing and solving problem. She/He will see an overview of the Artificial Intelligence languages for modeling problems and techniques for solving them. In particular the course is mainly focused on logical-declarative modeling and on constraint based solution search</p> <p>1.2 <i>Applying knowledge and understanding:</i> the student will be able to exploit the knowledge of the techniques and of the languages learnt for solving real-life problems, such as combinatorial problems, scheduling problem, automated reasoning problem etc. that are ubiquitous in industry.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 <i>Making judgements:</i> given the formal specifics of a problem, the student will have the capability of understanding if it is one that can be naturally solved with the techniques seen in the course. In particular, if the problem is NP complete, these techniques allow compact encodings and allow to exploit the "AI" embedded into the solvers for solutions search. Similarly, if the problem is a KR problem where several agents have to reason individually or together for reaching a</p>	
--	--	---	--

		<p>efficiente. Similmente, lo studente sarà in grado di modellare un problema che abbia a che fare con la rappresentazione della conoscenza in un sistema mono o multi agente (che cerca/cercano di ottenere un dato obiettivo data una situazione iniziale).</p> <p>2.2 Abilità comunicative: lo studente sarà in grado di padroneggiare la terminologia della parte "precisa" dell'intelligenza artificiale e sarà in grado di impiegarla opportunamente quando presenterà i propri risultati a specialisti e non.</p> <p>2.3 Capacità di apprendimento: lo studente, essendo venuto a conoscenza delle tecniche di base per la rappresentazione della conoscenza e per il ragionamento automatico, potrà impiegarle come punto di partenza per approfondire la disciplina o per la loro applicazione in discipline affini per il resto della sua vita. Il corso presenta prevalentemente materiale in lingua inglese e dunque ciò permette di approfondire la conoscenza della lingua straniera.</p>	<p>specified goal, the student will know how to model and solve it.</p> <p>2.2 Communication skills: the student will learn the precise terminology, the possibilities, and the intrinsic limits of the "exact" part of artificial intelligence and is able to use them properly when presenting his work even to non-specialists.</p> <p>2.3 Learning skills: the student will learn some of the "magic" that is inside artificial intelligence and can use this knowledge as a starting point for the study of development of the area and of its application in several fields in the remaining part of its life. Of course, being the course held in English, she/he will improve her/his language skills.</p>	
Recommender systems	ING-INF/05	<p>Obiettivi del corso sono:</p> <ul style="list-style-type: none"> - Conoscere i concetti e gli algoritmi fondamentali delle tecniche di Personalizzazione adattativa dei Contenuti WEB, dello User Modeling e dei Recommender Systems - riuscire a comprendere come le innovazioni del Web ne influenzano e ne influenzeranno l'utilizzo. - saper definire e progettare un modello utente. 	<p>The course objectives are:</p> <ul style="list-style-type: none"> - Knowing basic concepts and algorithms of adaptive personalization techniques, user modeling and recommender systems; - Understanding the relationship between Web personalization and the evolution of Web and Internet - Knowing how to specify and to design a user model 	

		<ul style="list-style-type: none"> - saper scegliere le tecniche di personalizzazione (adattativa) più adeguate. - saper analizzare un problema di accesso all'informazione personalizzato e proporre una possibile soluzione. <p>Lo studente dovrà:</p> <p>Capacità relative alle discipline</p> <p>3. <i>Conoscenza e capacità di comprensione:</i> acquisire specifiche conoscenze dei principali concetti e principi teorici delle tecniche di Personalizzazione dei Contenuti WEB. Conoscere e sapere utilizzare le tecniche di personalizzazione adattativa</p> <p>4. <i>Capacità di applicare conoscenza e comprensione:</i> saper analizzare e comprendere un algoritmo di personalizzazione adattativa, saper analizzare ed interpretare un problema di accesso personalizzato all'informazione ed applicare le conoscenze di cui sopra per affrontare specifiche problematiche applicative, progettare l'architettura logica di un sistema di personalizzazione adattativa per la soluzione di problemi reali.</p> <p>Capacità trasversali / soft skills</p> <p>Lo studente dovrà:</p> <p>4. <i>Autonomia di giudizio:</i> saper valutare i sistemi Web dal punto di vista dell'opportunità di utilizzare tecniche di personalizzazione adattativa e saper effettuare una scelta della tecnica ritenuta più adatta per la soluzione. Saper distinguere tra diverse soluzioni di</p>	<ul style="list-style-type: none"> - Knowing how to select the most adequate personalization techniques - Knowing how to analyze a personalized information access problem and how to propose a solution <p>The student will have to:</p> <p>Sector-specific skills</p> <p>3. <i>Knowledge and understanding:</i> acquiring specific knowledge of the main concepts and basic principles of Web Content Personalization. Knowing and exploiting techniques for adaptive personalization.</p> <p>4. <i>Applying knowledge and understanding:</i> knowing how to analyze and interpret an adaptive personalization algorithm, how to analyze an information access problem, how to apply the above mentioned knowledge in specific application context, how to design the logical architecture of a recommender system.</p> <p>Cross-sectoral skills/soft skills</p> <p>The student will have to:</p> <p>4. <i>Making judgments:</i> being able to independently evaluate the characteristics of the tools and methodologies to be applied in the various contexts of recommender and adaptive personalization systems.</p> <p>5. <i>Communication skills:</i> acquiring the ability to describe effectively and</p>	
--	--	---	--	--

		<p>personalizzazione adattativa e valutarne la validità.</p> <p>5. <i>Abilità comunicative</i>: saper illustrare con rigore logico e terminologico, a voce e per iscritto, questioni tecniche inerenti tecniche e sistemi di personalizzazione adattativa dei contenuti Web.</p> <p>6. <i>Capacità di apprendimento</i>: saper reperire e utilizzare strumenti bibliografici e informatici utili per l'approfondimento autonomo di problemi inerenti la personalizzazione adattativa dei contenuti Web.</p>	<p>through appropriate models the scenario of an adaptive personalized system, its benefits and risks</p> <p>6. <i>Learning skills</i>: being able to learn the overall functioning of recommender systems and their implications.</p>	
Ricerca operativa Operations research	MAT/09	<p>Obiettivi formativi specifici</p> <p>L'obiettivo culturale del corso è quello di presentare le principali metodologie modellistiche utilizzate nella risoluzione di problemi di ottimizzazione, cercando di distinguere fra problemi facili (polinomiali) e difficili (NP-hard). Per i primi, lo studente dovrebbe essere in grado di riconoscere un approccio della letteratura ed applicare il relativo algoritmo standard, adattandolo magari al caso specifico. Per i secondi, lo studente dovrebbe essere in grado di disegnare un modello di programmazione lineare intera, individuando vincoli e variabili, da risolvere poi con strumenti software standard per questo tipo di modelli.</p> <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e comprensione: Lo studente conosce i principali problemi di ottimizzazione NP-hard e polinomiali. E' in grado di trovare, se esiste, una semplice riduzione da un problema</p>	<p>Specific educational objectives (ENG)</p> <p>The goal of the course is to teach the main techniques for modeling and solving optimization problems, trying to distinguish between easy (i.e., polynomial) problems and hard problems (i.e., NP-hard). For the former ones, the student should be able to recognize a suitable approach from the literature and apply the corresponding standard algorithm, perhaps adapting it to the specific case. For the latter ones, the student should be able to write an integer LP model, introducing variables and constraints, to be then solved by using a standard software for this type of models.</p> <p>Sector-specific skills:</p> <p>1.1. Knowledge and understanding: The student knows the most important NP-hard and polynomial optimization</p>	<p>Nozioni di base di matematica discreta, teoria dei grafi, algebra lineare, algoritmi e programmazione. Basic notions of discrete mathematics, graph theory, linear algebra, algorithms and programming.</p>

	<p>NP-hard ad un altro. E' in grado di identificare un sottoproblema polinomiale noto e risolto in letteratura. Conosce l'arte di modellare un problema tramite la programmazione lineare intera. Conosce i fondamenti della programmazione matematica e i concetti di linearità e convessità di insiemi e funzioni. Conosce il funzionamento di base dei moderni solver di programmazione lineare intera nonché i fondamenti matematici alla base della PL e PLI.</p> <p>1.2 Capacità di applicare conoscenza e comprensione: Lo studente sa formulare un modello di PLI per risolvere un problema di ottimizzazione. Conosce gli algoritmi più efficienti in letteratura per la soluzione di problemi noti, e conosce semplici implementazioni di algoritmi efficaci. Riesce a valutare la complessità di un problema e l'efficacia di soluzioni alternative.</p> <p>Capacità trasversali /soft skills</p> <p>2.1. Autonomia di giudizio: Lo studente è in grado di percepire per quali problemi un approccio esatto può essere efficace e per quali si rendono necessari approcci euristici</p> <p>2.2. Abilità comunicative: lo studente conosce la terminologia tecnica della programmazione lineare, programmazione matematica, modellizzazione di problemi di ottimizzazione e descrizione critica di soluzioni alternative.</p> <p>2.3. Capacità di apprendimento: Lo studente acquisisce le basi del linguaggio matematico per</p>	<p>problems. He can find, if one exists, a simple reduction from an NP-hard problem to another. He can identify a known polynomial subproblem and pick its solution from the literature. He knows the art of modeling a problem as an integer linear programming instance. He knows the basics of mathematical programming, linearity and convexity of both sets and real functions. He knows the basic functioning of modern, state-of-the-art solvers for integer linear programming models as well as the theoretical foundations of both linear programming and integer programming.</p> <p>1.2. Applying knowledge and understanding: The student can formulate an ILP model for the solution of an optimization problem. He knows the most effective algorithms from the literature for the solution of famous known problems, and he can write simple implementation of many algorithms. He can assess the complexity of a problem and pick an effective solution among many alternatives.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements: The student is capable of making an educated guess about the effectiveness of an exact approach versus a heuristic for a given problem.</p>	
--	---	--	--

		<p>la formulazione di modelli di ottimizzazione. E' in grado, con uno sforzo minimale, di tradurli in input per solver commerciali non appena vengano fissate le regole sintattiche che definiscono gli input.</p>	<p>2.2 Communication skills: The student is familiar with the technical mathematical/linear programming jargon, the language of problem modeling and the critical assessment of alternative solutions.</p> <p>2.3 Learning skills: The student acquires the basics of the mathematical language for optimization problems modeling. He is capable of formulating a mathematical model and he should be able, with a minimal effort, to translate it into the input for a commercial solver once he is given the syntax rules defining the input format.</p>	
Semantica dei linguaggi di programmazione	INF/01	<p>Scopo del corso è fornire basi e strumenti per la definizione e lo studio della semantica dei linguaggi di programmazione, sia sequenziali (imperativi e funzionali) che concorrenti o distribuiti, questi ultimi aventi aspetti di comunicazione, sicurezza e mobilità tipiche del calcolo globale.</p> <p>Allo scopo vengono presentati una serie di linguaggi di programmazione con differenti caratteristiche e differenti livelli di complessità. Per ciascun linguaggio vengono definite le appropriate semantiche (denotazionali e/o operazionali strutturate), e gli strumenti formali per la verifica di proprietà di programmi.</p> <p>Capacità relative alla disciplina</p> <p>1.1 <i>Conoscenza e capacità di comprensione:</i> conoscere e comprendere le principali tecniche usate per descrivere formalmente la semantica dei linguaggi; sapere come vengono modellati i vari aspetti della computazione: higher-order,</p>	<p>The aim of the course is to provide the basics and tools for the definition and study of the semantics of programming languages, both sequential (imperative and functional) and concurrent or distributed, which have communication, security and mobility features typical of global computing.</p> <p>For this purpose, a series of programming languages with different features and different levels of complexity are presented.</p> <p>For each language, the appropriate semantic (denotational and / or operational), and the formal tools for program property verification are defined.</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding:</i> to know and understand the main techniques used to formally describe the semantics of</p>	

		<p>non-determinismo, concorrenza; conoscere le principali problematiche dei sistemi concorrenti e distribuiti.</p> <p><i>1.2 Conoscenza e capacità di comprensione applicate:</i> saper applicare tecniche per la verifica formale di proprietà di programmi o sistemi.</p> <p>Capacità trasversali/soft skills</p> <p><i>2.1 Autonomia di giudizio:</i> lo studente impara a essere cosciente delle ambiguità presenti nelle descrizioni informali, e individuare le tecniche di formalizzazione più opportune.</p> <p><i>2.2 Abilità comunicative:</i> lo studente impara a formalizzare e a descrivere in maniera rigorosa il comportamento dei vari costrutti di un linguaggio di programmazione.</p> <p><i>2.3 Capacità di apprendimento:</i> il corso fornisce le nozioni base di metodi formali necessarie per poter apprendere tecniche di verifica più sofisticate.</p>	<p>programming languages; to know how the various aspects of computation are modeled:</p> <p>higher-order, non-determinism, concurrence; to know the main issues of concurrent and distributed systems.</p> <p><i>1.2 Applying knowledge and understanding:</i> to know how to apply formal verification techniques for property of programs or systems.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>2.1. Making judgments:</i> students learn to be aware of the ambiguities present in informal descriptions, and identify the most appropriate formalization techniques.</p> <p><i>2.2 Communication skills:</i> students learn to formalize and describe behavior from the various constructs of a programming language.</p> <p><i>2.3 Learning skills:</i> the course provides the basic notions of formal methods needed to learn more sophisticated verification techniques.</p>	
<p>Sicurezza delle reti di calcolatori</p> <p>Computer Network Security</p>	INF/01	<p>Obiettivi</p> <p>L'obiettivo del corso è fornire agli studenti i metodi e le conoscenze per affrontare l'analisi, progettazione, implementazione e manutenzione dei servizi di sicurezza delle reti di calcolatori.</p> <p>In primo luogo, gli studenti imparano i concetti base della sicurezza di rete, quali asset, goal di sicurezza, vulnerabilità, minaccia, malware, attacchi, difese, modelli di sicurezza, ecc. In seguito, vengono analizzati i vari meccanismi per</p>	<p>The aim of the course is to provide students with the methods and knowledge to deal with the analysis, design, implementation and maintenance of services for computer network security. First, students learn the basic concepts of network security, such as assets, security goals, vulnerabilities, threats, malware, attacks, defenses, security models, etc. Subsequently, the various mechanisms for the implementation of effective defense</p>	Reti di calcolatori

		<p>l'implementazione di efficaci servizi di difesa, tra cui sistemi crittografici a chiave simmetrica e asimmetrica, autenticazione di messaggi, autenticazione di utenti e di entità, sistemi di prevenzione e/o di rilevamento delle intrusioni, piattaforme software e hardware fidate. Il corso è corredato da esempi pratici ed esercitazioni, con l'utilizzo di strumenti quali packet sniffer, honeypot, verificatori formali di protocolli.</p> <p>Al termine del corso gli studenti saranno in grado di progettare l'architettura di sistemi ed applicazioni sicure, e aggiornare autonomamente le proprie competenze nel settore, anche tramite risultati recenti della ricerca nell'area</p> <p>Capacità relative alle discipline</p> <p>1.1 Conoscenza e capacità di comprensione: lo studente acquisisce le conoscenze e capacità necessarie per analizzare e progettare i servizi di sicurezza di una rete di calcolatori, e comprendere le problematiche ad essa connesse. Le conoscenze coprono molteplici aspetti: crittografia, protocolli di sicurezza, firme digitali, autenticazione, analisi del rischio, piattaforme fidate, verifica formale.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: grazie ad una serie di esempi presentati a lezione e esercizi mirati alla risoluzione di casi pratici, lo studente acquisisce specifiche capacità di applicare a casi reali le conoscenze maturate sui vari aspetti della disciplina.</p>	<p>services are analyzed, including symmetric and asymmetric cryptographic systems, message authentication, user and entity authentication, intrusion prevention and/or detection systems, trusted software and hardware platforms. The course is accompanied by practical examples and exercises, with the use of tools such as packet sniffers, honeypots, formal protocol verifiers.</p> <p>At the end of the course students will be able to design the architecture of secure systems and applications, and independently update their skills in the sector, also through recent research results in the area.</p> <p>Sector-specific skills</p> <p>1.1 Knowledge and understanding: the student acquires the knowledge and skills necessary to analyze and design the security services of a computer network, and understand the problems related to it. The knowledge covers multiple aspects: cryptography, security protocols, digital signatures, authentication, risk analysis, trusted platforms, formal verification.</p> <p>1.2 Applied knowledge and understanding skills: thanks to a series of examples presented in class and exercises aimed at solving practical cases,</p>	
--	--	---	--	--

		<p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: lo studente acquisisce una capacità di valutazione critica dei diversi problemi di sicurezza di una rete di calcolatori e di come essi possono influire sul funzionamento di un sistema informativo aziendale, in diversi contesti d'uso e per diverse categorie di utenza.</p> <p>2.2 Abilità comunicative: lo studente impara a descrivere e progettare le problematiche di sicurezza di una rete di calcolatori in modo tecnicamente corretto ed usando la terminologia appropriata. Questo permette allo studente di inserirsi rapidamente in contesti professionali in cui questa terminologia tecnica è essenziale.</p> <p>2.3 Capacità di apprendimento: il corso fornisce le basi e gli strumenti che permettono allo studente di approfondire ed affrontare autonomamente problemi inerenti all'analisi, progettazione e realizzazione dei servizi di sicurezza di una rete di calcolatori.</p>	<p>the student acquires specific skills to apply the knowledge gained on various aspects of the discipline to real cases.</p> <p>Cross-sectoral skills / soft skills</p> <p>2.1 Autonomy of judgment: the student acquires a critical assessment ability of the various security problems of a computer network and how they can affect the functioning of a corporate information system, in different contexts and for different categories of users.</p> <p>2.2 Communication skills: the student learns to describe and plan the security problems of a computer network in a technically correct way and using the appropriate terminology. This allows the student to quickly enter professional contexts where this technical terminology is essential.</p> <p>2.3 Learning skills: the course provides the basics and tools that allow the student to investigate and independently deal with problems related to analysis, design and implementation of the security services of a computer network.</p>	
Sistemi distribuiti	INF/01	<p>Obiettivi</p> <p>L'obiettivo del corso è fornire agli studenti i metodi e le conoscenze per individuare e affrontare le questioni e i problemi che sorgono nella progettazione e realizzazione di sistemi distribuiti, come architetture orientate ai servizi, servizi cloud-based, sistemi peer-to-peer, ambienti collaborativi, eccetera.</p>	<p>The aim of the course is to provide students with the methods and knowledge to identify and address the issues and problems that arise in the design and implementation of distributed systems, such as service-oriented architectures, cloud-based services, peer-to-peer systems, collaborative environments, and so on.</p>	Reti di calcolatori

	<p>In primo luogo, gli studenti imparano i punti di forza intrinseci e le limitazioni di sistemi distribuiti rispetto ai sistemi sequenziali o fortemente accoppiati, e come sfruttare correttamente questi punti di forza. A tal fine, si studiano le architetture principali, i paradigmi di comunicazione basata su messaggi e i middleware utilizzati nei sistemi distribuiti. Poi, vengono introdotti Jolie e Erlang, due linguaggi specificamente progettati per la programmazione distribuita. Infine, si studiano gli algoritmi per la risoluzione di molti problemi comuni in sistemi distribuiti, come ad esempio il rilevamento dei fallimenti, il rilevamento dei deadlock, il consenso, la computazione peer-to-peer, e transazioni distribuite.</p> <p>Capacità relative alle discipline</p> <p>1.1 Conoscenza e capacità di comprensione: lo studente acquisisce le conoscenze e capacità necessarie per analizzare e progettare un sistema distribuito, e comprendere eventuali problematiche ad esso connesse. Le conoscenze coprono molteplici aspetti: architettura fisica e logica, vari modelli client/server, p2p, multi-tier; paradigmi di comunicazione; middleware a oggetti, a componenti, a servizi; linguaggi specifici per sistemi distribuiti; algoritmi per la risoluzione dei problemi ricorrenti dei sistemi distribuiti.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: grazie ad una serie di esempi presentati a lezione e esercizi mirati alla risoluzione di casi pratici, lo</p>	<p>First, students learn intrinsic strengths and limitations of distributed systems over sequential or strongly coupled systems, and how to properly leverage these strengths. To this end, the main architectures, the communication-based paradigms and the middleware used in distributed systems are studied. Then, Jolie and Erlang, two languages specifically designed for distributed programming, are introduced. Finally, the course presents the main algorithms to solve many common problems in distributed systems, such as failure detection, deadlock detection, peer-to-peer computing, and distributed transactions.</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding:</i> The student acquires the knowledge and skills needed to analyze and design a distributed system, and to understand related issues. This knowledge covers multiple aspects: physical and logical architectures, various client/server, p2p, multi-tier models; communication paradigms; middlewares oriented to objects, components, services; specific languages for distributed systems; algorithms for solving recurrent problems of distributed systems.</p> <p>1.2 <i>Applying knowledge and understanding:</i> By means of a series of examples presented in class and exercises aimed at solving practical cases, the student will</p>	
--	--	---	--

		<p>studente acquisisce specifiche capacità di applicare a casi reali le conoscenze maturate sui vari aspetti della disciplina.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: lo studente acquisisce una capacità di valutazione critica delle diverse caratteristiche di un sistema distribuito e di come esse possono influire sul funzionamento di un sistema informativo aziendale o di una applicazione distribuita, in diversi contesti d'uso e per diverse categorie di utenza.</p> <p>2.2 Abilità comunicative: lo studente impara a descrivere e progettare un sistema distribuito in modo tecnicamente corretto ed usando la terminologia appropriata. Questo permette allo studente di inserirsi rapidamente in contesti professionali in cui questa terminologia tecnica è essenziale.</p> <p>2.3 Capacità di apprendimento: il corso fornisce le basi e gli strumenti che permettono allo studente di approfondire ed affrontare autonomamente problemi inerenti all'analisi, progettazione e realizzazione di un sistema distribuito.</p>	<p>learn how to apply to real-life cases the knowledge about the various aspects of the discipline.</p> <p>Cross-sectoral skills / soft skills</p> <p>2.1. <i>Making judgments</i>: the student acquires the capability of assessing the different characteristics of a distributed system and how they can affect the functioning of a business information system or distributed application in different contexts and for different categories of users.</p> <p>2.2 <i>Communication Skills</i>: the student learns how to describe and design a distributed system in a technically correct manner and using the appropriate terminology. This allows the student to be quickly acquainted in professional contexts where this technical terminology is essential.</p> <p>2.3 <i>Learning Skills</i>: The course provides the basics and tools that enable the student to deepen and face independently issues related to the analysis, design and realization of a distributed system.</p>	
Sistemi informativi e data warehouse	ING-INF/05	<p>Obiettivo del corso è la definizione delle problematiche informatiche nell'ambito di un'organizzazione che intenda automatizzare i propri sistemi informativi. Si inquadrano le competenze del responsabile EDP e/o di chi sviluppa software per la gestione aziendale, sia a livello operativo che direzionale, analizzando le problematiche applicative che dovrà affrontare, con uno sguardo ai risvolti tecnologici ed ai rapporti di fornitura.</p>	<p>The course objective is to define IT issues within an organization, which has the goal to automate its information systems, both at the operational level and at the strategic management level. This includes the competence of EDP managers and/or of business management software developers, analyzing the applicative issues they will face, taking into</p>	

		<p>Lo studente dovrà:</p> <p>Capacità relative alle discipline</p> <p>5. <i>Conoscenza e capacità di comprensione:</i> conoscere i temi inerenti l'informatizzazione di una organizzazione complessa da un punto di vista organizzativo, di analisi dei costi e dei rischi, delle metodologie di progetto, dell'architettura hardware e software, delle funzioni dei sistemi ERP (Enterprise Resource Planning), dei sistemi di BI (Business Intelligence), di Performance Management e di KM (Knowledge Management).</p> <p>6. <i>Capacità di applicare conoscenza e comprensione:</i> organizzare l'informatica all'interno di un'organizzazione, analizzare le richieste dell'utenza, predisporre un capitolato/progetto per la fornitura, definire l'architettura hardware e software del sistema informativo, sviluppare specifiche applicazioni nell'ambito del sistema informativo aziendale, organizzare e sviluppare gli strumenti di business intelligence e performance management, reporting e supporto, gestire la contrattualistica di fornitura.</p> <p>Capacità trasversali / soft skills</p> <p>7. <i>Autonomia di giudizio:</i> essere in grado di valutare autonomamente quali siano le caratteristiche degli strumenti e delle metodologie da applicare nei diversi contesti di informatizzazione di dati e processi aziendali.</p>	<p>consideration also technology and supply relationships.</p> <p>The student will have to:</p> <p>Sector-specific skills</p> <p>5. <i>Knowledge and understanding:</i> knowing the various aspects related to the computerization of a complex organization from an organizational point of view, cost and risk analysis, project methodologies, hardware and software architecture, ERP (Enterprise Resource Planning) BI (Business Intelligence), Performance Management and KM (Knowledge Management) systems functionalities.</p> <p>6. <i>Ability to apply knowledge and understanding:</i> organizing IT within an organization, analyzing customer requests, setting up a supply specifications project, defining hardware and software architecture of the information system, developing specific applications for the information system, organizing and developing business intelligence and performance management, reporting and support tools, and managing supply contracts.</p> <p>Cross-sectoral skills/soft skills</p> <p>7. <i>Making judgments:</i> being able to independently evaluate the characteristics of the tools and methodologies to be applied in the various contexts of computer data and business processes definition.</p>	
--	--	---	--	--

		<p>8. <i>Abilità comunicative</i>: acquisire la capacità di descrivere in modo efficace e tramite adeguati modelli le necessità aziendali connesse alla trattazione e organizzazione di dati e processi aziendali.</p> <p>9. <i>Capacità di apprendimento</i>: essere in grado di apprendere il funzionamento complessivo dei sistemi informativi e le implicazioni di una corretta gestione di progetto, analisi dei rischi, modellazione dei dati e dei processi e della loro successiva elaborazione.</p>	<p>8. <i>Communicative skills</i>: acquiring the ability to describe effectively and through appropriate models the needs associated with processing and organization of business data and processes.</p> <p>9. <i>Learning skills</i>: being able to learn the overall functioning of information systems and the implications of proper project management, risk analysis, data and process modeling and their subsequent processing.</p>	
Statistica applicata	SECS-S/01	<p>Capacità relative alla disciplina <i>Conoscenza e capacità di comprensione</i>: conoscenza e comprensione delle principali procedure univariate e multivariate per sintetizzare i dati; conoscenza e comprensione dei principali modelli statistici e delle tecniche più importanti dell'apprendimento statistico, con particolare attenzione ai modelli di regressione e alle tecniche di analisi multivariata; conoscenza di almeno un software statistico. <i>Capacità di applicare conoscenza e comprensione</i>: comprensione dei metodi statistici come strumenti di ricerca utili in vari contesti applicati; capacità di usare la statistica descrittiva e inferenziale per sintetizzare informazioni, per analizzare e interpretare relazioni tra variabili e per test di ipotesi, acquisire abilità nell'utilizzazione di un software statistico.</p> <p>Capacità trasversali/soft skills <i>Autonomia di giudizio</i>: autonomia di giudizio nella scelta dei modelli e dei metodi statistici più</p>	<p>The course focuses on statistical methods for data analysis. The aim is to introduce the fundamental elements of statistical modelling and the basic concepts of statistical learning, with particular attention to regression models and multivariate data analysis techniques. These notions will be presented from an applied point of view and part of the course will take place in the computer lab, using the R statistical software. Course outline: 1) Introduction to statistics and data analysis; 2) Explorative data analysis; 3) A review of inference concepts; 4) Linear regression with a single predictor; 5) Towards multiple linear regression and logistic regression; 6) Predictive and classification methods; 7) Unsupervised methods (principal component analysis, cluster analysis).</p> <p>Sector-specific skills 1.1 <i>Knowledge and understanding</i></p>	

		<p>appropriati per analizzare uno specifico dataset e nell'interpretazione dei risultati sperimentali.</p> <p><i>Abilità comunicative:</i> abilità comunicative nel presentare in modo convincente e corretto un'analisi statistica, motivando i risultati ottenuti e giustificando la metodologia adottata.</p> <p><i>Capacità di apprendimento:</i> capacità di apprendimento utilizzando strumenti utili per riuscire a capire autonomamente i contenuti di un report statistico e per acquisire tecniche statistiche più avanzate.</p>	<p>Knowledge and understanding of univariate and multivariate descriptive statistics and of how to summarize and visualize data, of the basics in inferential statistics, of the fundamental elements of statistical modelling, of the basic concepts of statistical learning, focusing on regression models and multivariate data analysis techniques, and understanding of at least one statistical software for data analysis and statistical learning applications.</p> <p><i>1.2 Applying knowledge and understanding</i></p> <p>Understanding of statistical methods as useful instruments for research in economics and social sciences, ability to use descriptive and inferential statistics in order to summarize information, to analyze and interpret relationships between variables and to test hypotheses, ability to use at least one statistical software in order to develop simple data analysis.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>2.1 Making judgements</i></p> <p>Making judgements on the appropriate statistical models and methods to be used for analyzing a specific dataset and on the interpretation of the experimental results.</p> <p><i>2.2 Communication skills</i></p> <p>Communication skills in order to present a statistical analysis, including both the methodology and the final conclusions, in a consistent and convincing way.</p>	
--	--	--	---	--

			<p><i>2.3 Learning skills</i> Learning skills based on the prerequisites that are required for understanding autonomously a report</p>	
Verifica automatica dei sistemi: teoria e applicazioni	INF/01	<p>Obiettivi</p> <p>Il corso vuole presentare in modo sistematico metodi, formalismi e algoritmi per la specifica formale e la verifica automatica di sistemi reattivi complessi e alcune loro applicazioni nell'area dell'intelligenza artificiale (problemi di sintesi, planning e apprendimento) e della cybersecurity. Lo studente imparerà a padroneggiare strumenti formali di natura avanzata, sviluppati nei settori della logica matematica, della teoria degli automi e della teoria dei giochi, per l'analisi del comportamento e la verifica automatica di proprietà di sistemi complessi. Inoltre, avrà modo di conoscere e sperimentare alcuni degli ambienti di verifica più significativi disponibili in ambito informatico. Ampio spazio verrà dedicato alla teoria degli automi a stati finiti e alle loro varianti operanti su oggetti infiniti (parole e alberi) e alle logiche temporali comunemente utilizzate (LTL, CTL e CTL*). Particolare attenzione verrà riservata ai risultati relativi all'equivalenza espressiva tra classi di automi e sistemi logici. Verrà, inoltre, illustrato il</p>	<p>Objectives</p> <p>The course aims at providing a systematic account of computer science methods, formalisms, and algorithms for the formal specification and automatic verification of complex reactive systems and some applications of them in the area of artificial intelligence (synthesis problems, planning, learning) and cybersecurity. The student will learn how to master advanced formal tools, developed in the fields of mathematical logic, automata theory, and logical game theory, to be used for the analysis of the behavior of complex systems and the automatic verification of their relevant properties. In addition, he/she will have the possibility of analyzing and practicing some of the most significant verification frameworks developed in computer science. A prominent role is assigned to the theory of finite-state automata and their variants operating on infinite objects (words and trees) and to the most commonly</p>	

	<p>possibile utilizzo della teoria logica dei giochi nella verifica formale.</p> <p>Dal punto di vista algoritmico, con riferimento ai modelli computazionali e ai formalismi di specifica presi in considerazione, verranno studiati in dettaglio gli algoritmi per la verifica della consistenza delle specifiche e della correttezza dei modelli. Un ruolo centrale verrà assegnato agli algoritmi di model checking, che consentono di validare il comportamento di un sistema hardware o software, descritto formalmente attraverso un modello matematico appropriato (ad esempio, un automa), rispetto alle proprietà attese del sistema, specificate mediante formule logiche (ad esempio, formule di CTL). In particolare, verranno illustrate le soluzioni proposte per migliorare le prestazioni degli algoritmi di model checking (OBDD e model checking simbolico, partial order reduction, bounded model checking). Verranno, inoltre, presentati alcuni degli ambienti di verifica più rappresentativi (nuSVM, SPIN, UPPAAL). Nell'ultima parte del corso verranno introdotti alcuni temi di carattere più avanzato, quali il problema della sintesi e la verifica di sistemi a stati infiniti.</p>	<p>used temporal logics (LTL, CTL, and CTL*). A special attention is given to the expressive equivalence between classes of automata and logical systems. The possible role of the logical theory of games in formal verification is illustrated as well.</p> <p>From the algorithmic point of view, on the basis of the considered computational models and specification formalisms, the course studies in detail the main algorithms for checking the consistency of specifications (satisfiability checking) and model checking. A special emphasis is given to model checking algorithms, which make it possible to validate the behavior of hardware and software systems, described by means of a suitable mathematical model, e.g., an automaton, with respect to their formal requirements, specified by means of logical formulas, e.g., a CTL formula. In particular, the course illustrates the main solutions that have been proposed in the literature to improve the performance of model checking algorithms (OBDD and symbolic model checking, partial order reduction, bounded model checking). In addition, it presents some of the most significant verification</p>	
--	---	--	--

	<p>Capacità relative alle discipline</p> <p>1.1 Conoscenza e capacità di comprensione:</p> <p>Il corso vuole fornire allo studente un insieme di nozioni e di strumenti concettuali che gli consentano di modellare in modo appropriato sistemi complessi e di analizzarne in modo sistematico (algoritmico) le proprietà fondamentali. Per promuoverne la capacità di comprensione, verranno illustrati approcci alternativi ai problemi considerati, dei quali verranno evidenziate somiglianze e peculiarità.</p> <p>1.2 Capacità di applicare conoscenza e comprensione:</p> <p>Lo studente avrà modo di verificare l'effettiva comprensione degli argomenti del corso su un insieme rappresentativo di problemi, alcuni abbastanza semplici, altri più difficili, che gli verranno assegnati come homework. La discussione con ciascuno studente delle soluzioni agli esercizi proposti è parte integrante del corso.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio:</p>	<p>frameworks (nuSVM, SPIN, UPPAAL). In the last part of the course, some advanced topics are briefly introduced such as the synthesis problem and infinite state system verification.</p> <p>Sector-specific skills</p> <p>1.1 Knowledge and understanding:</p> <p>The course aims at providing the student with a set of notions and conceptual tools that allow him/her to properly model complex systems and to analyze their fundamental properties in a systematic (algorithmic) way. In order to improve his/her understanding of the main contents of the course, alternative approaches to the considered problems will be illustrated, pointing out their similarities and peculiarities.</p> <p>1.2 Applying knowledge and understanding:</p> <p>The student will check his/her actual understanding of the contents of the course on a representative set of problems, some of them relatively easy to solve, others definitely more complex, which will be</p>	
--	--	---	--

		<p>Lo studente dovrà produrre una relazione scritta, di natura tecnica, su un argomento già affrontato in letteratura legato alle tematiche del corso. L'argomento può essere uno degli argomenti suggeriti dal docente durante il corso o può essere proposto in modo autonomo dallo studente (in questo secondo caso, è necessaria l'approvazione da parte del docente). L'argomento può essere sia di natura teorica sia di natura pratica. Tale attività vuole promuovere le capacità di analisi critica e di elaborazione autonoma dello studente.</p> <p>2.2 Abilità comunicative:</p> <p>La stesura della relazione di natura tecnica sarà seguita da una presentazione orale dei suoi contenuti principali, alla quale sono invitati tutti gli studenti che hanno seguito il corso. Tale attività intende promuovere le abilità comunicative dello studente, in particolare le capacità di esposizione scritta e orale di contenuti avanzati di natura tecnica.</p> <p>2.3 Capacità di apprendimento:</p> <p>La natura degli argomenti trattati nel corso, le modalità di organizzazione e presentazione dei diversi</p>	<p>assigned to him/her as homework. The discussion with each student of his/her solutions to the proposed problems is an essential part of the course.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements:</p> <p>The student must produce a written technical report on a topic from the literature related to the contents of the course. The topic can be either one of the topics suggested by the professor during the course or autonomously proposed by the student (in the latter case, the approval by the professor is requested). Moreover, the topic can be either a theoretical study or a more practical subject. Such an activity aims at promoting some fundamental abilities of the student, in particular, critical analysis and autonomous elaboration of a topic.</p> <p>2.2 Communication skills:</p> <p>The writing of the technical report will be followed by an oral presentation of its main contents. All students attending the course will be invited at such a presentation. Such an activity aims at</p>	
--	--	--	--	--

		<p>contenuti e le modalità di verifica della loro effettiva comprensione sono finalizzati alla crescita delle capacità di apprendimento degli studenti.</p>	<p>enhancing the communication abilities of the student, in particular his/her ability to illustrate, in written and verbal forms, advanced contents of technical nature.</p> <p>2.3 Learning skills:</p> <p>The nature of the topics addressed by the course, their organization and presentation modalities and the way in which the actual understanding by the student is checked aim at increasing the learning abilities of the students.</p>	
Video game programming	INF/01	<p>L'obiettivo del corso è introdurre i principi, le metodologie e gli strumenti applicativi per lo sviluppo di video game, inclusi i serious game, i video game per i dispositivi mobili e quelli per la realtà virtuale. Il corso accompagna l'introduzione dei concetti teorici e delle metodologie utilizzabili per programmare i diversi aspetti di un video game (ad esempio, l'interfaccia utente e l'intelligenza artificiale dei personaggi non giocanti) con l'applicazione pratica dei concetti e delle metodologie mediante l'utilizzo di uno dei game engine più diffusi. Il corso introduce anche alcuni concetti base sulla ottimizzazione e la valutazione dei video game (ad esempio, la game analytics), utili per migliorare la programmazione.</p> <p>Conoscenza e capacità di comprensione: gli studenti acquisiscono conoscenze specifiche sui diversi aspetti relativi alla programmazione di</p>	<p>The aim of the course is to introduce the principles, methodologies and application tools for the development of video games, including serious games, video games for mobile devices and those for virtual reality. The course introduces the theoretical concepts and the methodologies that can be used to program the different aspects of a video game (for example, the user interface and the artificial intelligence of non-playing characters), and shows the practical application of these concepts and methodologies by using one of the most popular game engines. The course also introduces some basic concepts concerning the optimization and evaluation of video games (for example, game analytics), useful for improving programming.</p>	<p>L'obiettivo del corso è introdurre i principi, le metodologie e gli strumenti applicativi per lo sviluppo di video game, inclusi i serious game, i video game per i dispositivi mobili e quelli per la realtà virtuale. Il corso accompagna l'introduzione dei concetti teorici e delle metodologie utilizzabili per programmare i diversi aspetti di un video game (ad esempio, l'interfaccia utente e l'intelligenza artificiale</p>

		<p>video game. Inoltre, imparano a scegliere tra le diverse metodologie proposte per ciascuno degli aspetti considerati, a seconda della piattaforma e degli utenti target del video game da programmare.</p> <p>Conoscenza e capacità di comprensione applicate: gli studenti acquisiscono competenze specifiche per applicare le conoscenze alla programmazione pratica di un video game, utilizzando uno dei game engine più diffusi.</p> <p>Autonomia di giudizio: gli studenti acquisiscono la capacità di valutare, sulla base di dati oggettivi, le prestazioni di un video game e di capire come le scelte di sviluppo possano influire sull'esperienza di gioco.</p> <p>Abilità comunicative: gli studenti imparano la terminologia appropriata per descrivere i vari aspetti della programmazione di un video game. Inoltre, apprendono come comunicare contenuti importanti, come informazioni relative alla salute e alla sicurezza delle persone, attraverso un video game.</p> <p>Capacità di apprendere: il corso fornisce le conoscenze e gli strumenti applicativi di base che consentono allo studente di approfondire e affrontare autonomamente questioni più complesse relative alla programmazione di video game.</p>	<p>Knowledge and understanding: students acquire specific knowledge about the different aspects related to video game programming. They also learn to choose from the different methodologies proposed for each of the considered aspects, depending on the platform and the target users of the video game to program.</p> <p>Applying knowledge and understanding: students acquire specific skills to apply knowledge to the practical programming of a video game, using one of the most popular game engines.</p> <p>Making judgements: students acquire the ability to evaluate the performance of a video game based on objective data and to understand how development choices can affect the gaming experience.</p> <p>Communication skills: students learn the appropriate terminology to describe the various aspects related to video game programming. Moreover, they learn how to communicate serious content, such as information about health and safety of people, by means of a video game.</p> <p>Learning skills: The course provides the basic knowledge and application tools that allow the student to deepen and autonomously address more complex</p>	<p>dei personaggi non giocatori) con l'applicazione pratica dei concetti e delle metodologie mediante l'utilizzo di uno dei game engine più diffusi. Il corso introduce anche alcuni concetti base sulla ottimizzazione e la valutazione dei video game (ad esempio, la game analytics), utili per migliorare la programmazione.</p> <p>Conoscenza e capacità di comprensione: gli studenti acquisiscono conoscenze specifiche sui diversi aspetti relativi alla programmazione di video game. Inoltre, imparano a scegliere tra le diverse metodologie proposte per ciascuno degli aspetti considerati, a seconda della piattaforma e degli utenti target del video game da programmare.</p>
--	--	---	---	---

			issues related to video game programming.	<p>Conoscenza e capacità di comprensione applicate: gli studenti acquisiscono competenze specifiche per applicare le conoscenze alla programmazione pratica di un video game, utilizzando uno dei game engine più diffusi.</p> <p>Autonomia di giudizio: gli studenti acquisiscono la capacità di valutare, sulla base di dati oggettivi, le prestazioni di un video game e di capire come le scelte di sviluppo possano influire sull'esperienza di gioco.</p> <p>Abilità comunicative: gli studenti imparano la terminologia appropriata per descrivere i vari aspetti della programmazione di un video game. Inoltre,</p>
--	--	--	---	--

				<p>apprendono come comunicare contenuti importanti, come informazioni relative alla salute e alla sicurezza delle persone, attraverso un video game.</p> <p>Capacità di apprendere: il corso fornisce le conoscenze e gli strumenti applicativi di base che consentono allo studente di approfondire e affrontare autonomamente questioni più complesse relative alla programmazione di video game.</p>
Virtual reality and persuasive user experience	INF/01	<p>L'obiettivo del corso è introdurre i principi, le metodologie e le applicazioni delle aree in rapida crescita della realtà virtuale e delle tecnologie persuasive, con un'attenzione approfondita su come gli utenti ne fanno esperienza (User Experience). La parte teorica del percorso include la percezione della realtà, i modelli psicologici di persuasione e i fattori che li riguardano. La parte metodologica del corso si occupa delle diverse scelte progettuali che devono essere intraprese per creare sistemi interattivi e esperienze coinvolgenti e</p>	<p>The aim of the course is to introduce the principles, methodologies, and applications of the rapidly growing areas of Virtual Reality and Persuasive Technologies, with a comprehensive focus on how users experience them (User Experience). The theoretical part of the course includes human perception of reality, psychological models of persuasion, and the factors that affect them. The methodological part of the course deals with the different design</p>	

	<p>persuasive, nonché con metriche e metodi per valutare la loro efficacia. La parte applicativa del corso presenta le diverse categorie di applicazioni della realtà virtuale, dei serious game e delle tecnologie persuasive, tra cui diversi casi di studio. Il corso include compiti pratici che consentono agli studenti di applicare le tecniche di valutazione apprese nel corso a casi di studio reali.</p> <p>Capacità relative alla disciplina</p> <p>1.1 <i>Conoscenza e comprensione</i>: gli studenti acquisiscono conoscenze multidisciplinari specifiche sulla realtà virtuale, sulla tecnologia persuasiva e sulla user experience. Inoltre, imparano a scegliere tra diverse tecniche per la progettazione di esperienze virtuali e persuasive, a seconda degli obiettivi dell'applicazione, del suo contesto di utilizzo e del suo utente target.</p> <p>1.2 <i>Capacità di applicare conoscenza e comprensione</i>: attraverso una serie di studi di casi e di compiti, gli studenti acquisiscono competenze specifiche per applicare la conoscenza della disciplina ai vari aspetti dei progetti reali riguardanti la realtà virtuale e le esperienze persuasive negli utenti.</p> <p>Capacità trasversali/Soft skills</p> <p>2.1. <i>Autonomia di giudizio</i>: gli studenti acquisiscono la capacità di valutare criticamente le diverse caratteristiche della realtà virtuale e della tecnologia persuasiva e come ogni scelta di progettazione può influire positivamente o negativamente sull'efficacia dell'esperienza utente in diversi contesti di utilizzo e per diverse categorie di utenti.</p>	<p>choices that have to be taken to create engaging and persuasive interactive systems and virtual experiences, as well as with the metrics and methods to evaluate their effectiveness. The application part of the course presents the different categories of applications of virtual reality, serious games, and persuasive technologies, including several real-world case studies. The course includes practical assignments that allow students to apply the evaluation techniques learned in the course to real-world case studies.</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding</i>: Students acquire specific multidisciplinary knowledge about virtual reality, persuasive technology, and user experience. They also learn to choose from various techniques for the design of virtual and persuasive experiences, depending on the objectives of the application, its context of use, and its target user.</p> <p>1.2 <i>Applied knowledge and understanding</i>: Through a series of case studies and assignments, students acquire specific skills to apply knowledge of the discipline to the various aspects of real-world projects concerning virtual reality and persuasive user experiences.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1. <i>Making judgments</i>: Students acquire the ability to critically evaluate the different features of virtual reality and persuasive technology, and how each</p>	
--	--	---	--

		<p>2.2 <i>Capacità di comunicazione</i>: Gli studenti imparano a descrivere la realtà virtuale e le esperienze persuasive dell'utente in modo tecnico corretto e utilizzando la terminologia appropriata. Il corso inoltre dedica diverse lezioni al tema della comunicazione persuasiva e tale conoscenza può essere utilizzata anche per la comunicazione interpersonale.</p> <p>2.3 <i>Capacità di apprendimento</i>: Il corso fornisce le conoscenze e gli strumenti che consentono allo studente di approfondire e affrontare autonomamente questioni relative alla progettazione e valutazione della realtà virtuale e alle esperienze persuasive negli utenti.</p>	<p>design choice can positively or negatively affect the effectiveness of the user experience in different contexts of use and for different categories of users.</p> <p>2.2 <i>Communication Skills</i>: Students learn to describe virtual reality and persuasive user experiences in a technically correct way and using the appropriate terminology. The course also devotes several lessons to the topic of persuasive communication, and such knowledge can be used also for interpersonal communication.</p> <p>2.3 <i>Learning skills</i>: The course provides the knowledge and tools that enable the student to deepen and address autonomously issues related to the design and evaluation of virtual reality and persuasive user experiences.</p>	
Web semantico	ING-INF/05	<p>Obiettivi formativi specifici</p> <p><i>Il progetto internazionale del Web Semantico</i></p> <p>Principi; finalità; aspetti concettuali;</p> <p>Cenni alle architetture per le applicazioni Web</p> <p><i>Modelli e linguaggi del Web Semantico</i></p> <p>Richiami a XML; limitazioni; XML Schema;</p> <p>Il modello dei dati RDF; sintassi; esercitazioni;</p> <p>RDFS e la modellazione concettuale di un dominio;</p> <p>Una semantica assiomatica per RDF;</p> <p>Il linguaggio di interrogazione SPARQL;</p> <p>esercitazioni</p> <p><i>Ontologie e Ontology Web Language (OWL)</i></p> <p>Rappresentazioni della conoscenza e Ontologie;</p> <p>I linguaggi OWL, OWL2; estensioni e profili</p>	<p>Specific Training Goals</p> <p><i>The International Semantic Web Initiative</i></p> <p>Basic principles; goals; conceptual issues</p> <p>Hints on the application architectures for the SW</p> <p><i>Formal Models and Languages of the Semantic Web</i></p> <p>The XML language and its limitations;</p> <p>XML Schema;</p> <p>The RDF data model; multiple syntaxes and formal rules;</p> <p>RDFS and domain modelling;</p> <p>An axiomatic syntax for RDF;</p> <p>The SPARQL query language: syntax and examples.</p>	

	<p>Architetture di sistemi applicativi: esempi Lo/la studente/essa dovrà:</p> <p>Capacità relative alle discipline</p> <p>1.1 <i>Conoscenza e capacità di comprensione.</i> Conoscere le motivazioni, le finalità, lo schema concettuale del WS. Conoscere la sintassi dei linguaggi XML, XML Schema. Conoscere i criteri alla base della modellazione RDF. Conoscere il linguaggio RDFS. Conoscere il linguaggio di interrogazione SPARQL. Conoscere principi e limitazioni dei linguaggi OWL, OWL2 e relative estensioni. Conoscere elementi delle tecnologie più usate nell'ambito del WS: ad esempio, l'ambiente APACHE-JENA.</p> <p>1.2 <i>Capacità di applicare conoscenza e comprensione.</i> Saper progettare un modello concettuale di un dominio circoscritto adottando come approccio le reti semantiche. Saper implementare un modello concettuale in termini di triple RDF. Saper implementare il modello secondo uno dei linguaggi OWL2 in una delle sintassi attualmente documentate. Saper operare in laboratorio usando l'ambiente APACHE-JENA.</p> <p>Capacità trasversali / soft skills</p> <p>2.1 <i>Autonomia di giudizio.</i> Saper analizzare un modello di dominio di interesse per il WS. Saper individuare formalismi di modellazione e linguaggi di implementazione adeguati per lo sviluppo di un'applicazione WS. Saper operare in un ambiente software e selezionare le librerie per la realizzazione pratica di un progetto.</p> <p>2.2 <i>Abilità comunicative.</i> Essere in grado di motivare per scritto, in un documento tecnico, le scelte di modellazione concettuale e</p>	<p><i>Ontologies and the Ontology Web Language (OWL)</i></p> <p>Knowledge Representations and Ontologies</p> <p>The languages OWL, OWL2; extensions, profiles</p> <p>Application architectures and examples.</p> <p>The student is requested to:</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding.</i> Be acquainted with the conceptual scheme of the SW initiative. Be familiar with the XML, XMLS languages. Be familiar with the basic principles of the RDF data model. Be acquainted with the language RDFS as well as the query language SPARQL. Know the syntax of OWL, OWL2, with extensions and limitations. Be acquainted with a popular SW technology: the APACHE JENA development environment.</p> <p>1.2 <i>Applying knowledge and understanding.</i> Be able to work out a domain conceptual model adopting semantic networks. Capability to implement a conceptual model in terms of RDF triples. Capability to implement a domain model according to an OWL2 syntax. Affording the software development work using APACHE-JENA.</p> <p>Cross-sectoral/soft skills</p> <p>2.1 <i>Making judgements.</i> Capabilities to critically analyze a restricted knowledge domain. Capabilities to single-out appropriate formal languages to develop a SW application. Capabilities to work in the</p>	
--	---	---	--

		<p>implementative effettuate nell'ambito di un progetto per il WS. Capacità di apprendimento: saper approfondire in modo autonomo le caratteristiche di un problema di modellazione concettuale nell'ambito del WS e relativamente alle tematiche trattate nel corso. Saper scrivere il software in termini di triple RDF. Saper affrontare lo studio di aggiornamenti e nuove soluzioni per il WS: standard tecnologici, ambienti di sviluppo.</p>	<p>realm of a software environment, and select the appropriate tools for the implementation of a WS project.</p> <p><i>2.2 Communication skills.</i> Be able to justify with a technical report the design choices taken on both conceptual and practical grounds. Learning abilities: to gain autonomously a deeper knowledge into a conceptual modelling effort, with respect to the topics addressed during the lectures. Be able to write and document a software program in terms of RDF triples. Be able to address novel technological solutions and standards for SW projects.</p>	
--	--	---	--	--