

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
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 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</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 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</p>	

	<p>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). 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 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> 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</p>	<p>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> 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>1.2 Applying knowledge and understanding</i> Students learn new declarative languages and learn to write code in the various</p>	
--	--	--	--

	<p>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> 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> Lo studente impara l'esatto significato dei termini usati nella programmazione e nella implementazione dei linguaggi. 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> 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>paradigms by using appropriate styles and techniques. The 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>2.1 Making judgements</i> 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><i>2.2 Communication skills</i> The student learns the exact meaning of the terms used in programming language coding and implementation. Through the group project activity, students improve their communicative and interaction skills.</p> <p><i>2.3 Learning skills</i> The knowledge acquired allows students to move with ease among the myriad of programming languages, and to learn new</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.	languages in a short time by quickly framing new language features. Thanks to the interaction with the other members of the project team, students learn to evaluate their degree of learning by comparing with others.	
Advanced Human-Computer Interaction	INF/01	<p>Il corso prevede una serie di lezioni frontali corredate da specifici esempi pratici ed applicativi concernenti il tema delle interfacce grafiche avanzate.</p> <p>Il corso mira a far acquisire allo studente le seguenti abilità:</p> <ul style="list-style-type: none"> - conoscere i più recenti sviluppi nel settore delle interfacce grafiche e le loro più importanti ricadute applicative - conoscere le principali tecniche di interazione avanzata con interfacce grafiche (ad esempio, app mobili o strumenti di bordo degli autoveicoli) - saper scegliere la tecnica più appropriata di visualizzazione delle informazioni ai fini di progetto di interfacce ed applicazioni visuali in diversi domini applicativi - saper inserire ed integrare nel progetto di applicazioni software complesse le considerazioni tecniche specifiche atte ad ottenere un'interazione con l'utente ed una visualizzazione delle informazioni ottimale <p>Si indicano i risultati di apprendimento attesi come riportati nei seguenti Descrittori di Dublino.</p> <p>Capacità relative alle discipline:</p> <ul style="list-style-type: none"> - <i>Conoscenza e comprensione</i> <p>Lo/la studente/essa acquisisce durante l'insegnamento di Advanced Human-Computer</p>	<p>The course includes a series of frontal lessons along with specific practical and applied examples regarding the topic of advanced graphics interfaces.</p> <p>The course aims for the student to acquire the following skills:</p> <ul style="list-style-type: none"> - know the most recent developments in the graphics interface industry and the resulting most important application impacts - know the main advanced interaction techniques with graphical interfaces (for example, mobile apps or on-board vehicle systems) - know how to choose the most appropriate visualization technique of information for project interfaces and visual applications in different application domains - know how to insert and integrate into complex software application projects the specific technical considerations designed to interact with the user and optimize information visualization <p>The expected learning outcomes are outlined as reported in the following Dublin descriptors.</p> <p>Sector-specific skills:</p>	

	<p>Interaction specifiche conoscenze dei principali concetti e principi teorici dell'interazione fra utente ed interfacce grafiche avanzate.</p> <p>Lo/la studente/essa impara inoltre a scegliere fra diverse tecniche di interazione impiegabili in un'interfaccia avanzata e a considerare variabili specifiche nelle valutazioni di tali interfacce sugli utenti</p> <p><i>- Capacità di applicare conoscenza e comprensione</i></p> <p>Lo/la studente/essa acquisisce durante l'insegnamento di Advanced Human-Computer Interaction specifiche capacità di applicare a casi reali la comprensione e le conoscenze maturate su visualizzazione delle informazioni..</p> <p>Lo studente sarà in grado di applicare la comprensione e le conoscenze acquisite alla realizzazione di progetti complessi nel settore delle interfacce grafiche avanzate.</p> <p>Capacità trasversali /soft skills</p> <p><i>- Autonomia di giudizio</i></p> <p>Lo/la studente/essa acquisirà con l'insegnamento di Advanced Human-Computer Interaction la capacità di valutare criticamente le caratteristiche che possono influire positivamente o negativamente sull'utente per ciò che concerne la visualizzazione delle informazioni e l'interazione con esse.</p> <p>Acquisirà inoltre la capacità di giudizio necessaria per valutare l'appropriatezza delle diverse tecniche apprese nell'insegnamento in diversi contesti applicativi.</p> <p><i>- Abilità comunicative</i></p> <p>Lo/la studente/essa acquisirà durante</p>	<p><i>- Knowledge and understanding</i></p> <p>During the teaching of Advanced Human-Computer Interaction the student acquires specific knowledge of the main concepts and theoretical principles of user interaction and advanced graphical interfaces. The student also learns to choose from different interaction techniques that can be used in an advanced interface and to consider specific variables in the evaluation of such user interfaces.</p> <p><i>- Applying knowledge and understanding</i></p> <p>During the teaching of Advanced Human-Computer Interaction the student acquires specific skills enabling him/her to apply the understanding and knowledge of information visualization to real cases. The student will be able to apply the understanding and knowledge gained to create complex projects in the field of advanced graphic interfaces.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>- Making judgements</i></p> <p>With the teaching of Advanced Human-Computer Interaction the student will acquire the ability to critically evaluate the features that can positively or negatively affect the user with regard to information visualization and interaction with it. He/she will also acquire the necessary judgment skills needed to evaluate the appropriateness of the different techniques in different application contexts that will be taught in lesson.</p>	
--	---	---	--

		<p>l'insegnamento di Advanced Human-Computer Interaction la conoscenza dei fattori principali che rendono efficace la comunicazione attraverso visualizzazione delle informazioni e potrà quindi riusare tali conoscenze anche nell'ambito della comunicazione interpersonale e professionale. Inoltre, l'insegnamento prevede la redazione di una relazione tecnica su attività di progetto o valutazione realizzate dallo studente.</p> <p>- <i>Capacità di apprendimento</i> Lo/la studente/essa acquisirà inoltre specifiche capacità trasversali relative all'abilità di identificare relazioni multidisciplinari fra aspetti tecnici (progettazione e sviluppo di interfacce grafiche avanzate, sistemi mobili e di realtà virtuale) e psicologici (teorie e modelli della percezione, dell'attenzione e dell'apprendimento).</p>	<p>- <i>Communication skills</i> During the teaching of Advanced Human-Computer Interaction the student will acquire knowledge of the main factors that make communication effective through information visualization and can re-use this knowledge in his/her interpersonal and professional communication. In addition, the course involves a technical report on project activities or assessment written by the student.</p> <p>- <i>Learning skills</i> The student will also acquire specific transversal skills related to the ability to identify multidisciplinary relationships between technical aspects (design and development of advanced graphical interfaces, mobile systems and virtual reality) and psychological ones (theories and models of perception, of attention and learning).</p>	
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> 	<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> 	

	<ul style="list-style-type: none"> ○ <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> ● 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>	<ul style="list-style-type: none"> ○ <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"> ○ <i>Randomness and Algoritmica</i> ○ <i>Algorithms and basic techniques</i> ○ <i>Rabin-Karp algorithm</i> ○ <i>Universal Hashing</i> ● 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>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> <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>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 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	Il corso descrive algoritmi, metodi e tecniche computazionali ampiamente utilizzati	The course focuses on algorithms, methods and computational techniques of	

	<p>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><i>1.1 Conoscenze e capacità di comprensione</i> Valutare la complessità di un algoritmo numerico e collocarla nell'ambito dei limiti noti per il problema. Saper distinguere fra le diverse 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><i>1.2 Conoscenze e capacità di comprensione applicate</i> 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><i>2.1 Autonomia di giudizio</i> Valutare criticamente un algoritmo alla luce dei risultati noti di complessità, paragonandolo ad altri algoritmi per lo stesso problema.</p> <p><i>2.2 Abilità comunicative</i> Descrivere le principali caratteristiche di un algoritmo numerico contestualizzandolo ed utilizzando un'appropriata terminologia.</p> <p><i>2.3 Capacità di apprendimento</i> Leggere libri o articoli in riviste specializzate per l'approfondimento autonomo circa i temi affrontati nel corso.</p>	<p>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><i>1.1 Knowledge and understanding</i> Evaluate the complexity of a numerical algorithm in connection with the known bounds for the problem. Recognize the 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><i>1.2 Applying knowledge and understanding</i> 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><i>2.1 Making Judgments</i> Evaluate an algorithm in the light of known complexity results, comparing it to other algorithms for the same problem.</p> <p><i>2.2 Communications skills</i> Describe the main features of a numerical algorithm by placing it in the right perspective and using appropriate terminology.</p> <p><i>2.3 Learning skills</i> Read books or articles in specialized journals in order to deepen the topics studied in the course.</p>	
--	--	---	--

<p>Analisi e verifica mediante interpretazione astratta</p>	<p>INF/01</p>	<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>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>1.1 Conoscenza e capacità di comprensione: 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</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>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>1.1 Knowledge and understanding 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>si possono costruire mediante Interpretazione Astratta.</p> <p>1.2 Conoscenza e capacità di comprensione applicate: 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 le conoscenze teoriche presentate durante le lezioni frontali.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 Autonomia di giudizio: 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>2.2 Abilità comunicative: Lo studente impara l'esatto significato dei termini usati nell'analisi e nella verifica dei sistemi software.</p> <p>2.3 Capacità di apprendere: Gli esercizi assegnati durante il corso e la loro correzione offrono allo studente gli strumenti</p>	<p>1.2 Applying knowledge and understanding 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>Cross-sectoral skills/soft skills</p> <p>2.1 Making judgements 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>2.2 Communication skills The student learns the exact meaning of the terms used in the fields of analysis and verification of software systems.</p> <p>2.3 Learning skills The exercises assigned during the course and their correction offer students the tools they need to evaluate their level of learning.</p>	
--	--	---	--

		necessari a valutare il proprio grado di apprendimento.		
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:</p> <p>avranno appreso i fondamenti della psicologia della percezione uditiva e tattile;</p> <p>conosceranno la natura di un segnale e di un sistema a tempo continuo e discreto;</p> <p>sapranno riconoscere e valutare lo spettro di un segnale a tempo discreto;</p> <p>saranno a conoscenza delle tecniche tradizionali per la sintesi del suono;</p> <p>avranno nozioni sufficienti per valutare e progettare semplici sistemi software e hardware per la sintesi di suoni e vibrazioni;</p> <p>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:</p> <p>Conoscenza e comprensione</p> <p>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></p> <p>Attraverso la soluzione di un numero sufficiente di esercizi, gli studenti comprendono la difficoltà</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 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>The expected learning outcomes are, as reflected in the following Dublin Descriptors.</p> <p>Sector-specific skills</p> <p><i>Knowledge and understanding:</i></p> <p>Students acquire basic knowledge about sound and vibrations in their fundamental components of perception, mathematics, and operation. The same knowledge are</p>	

	<p>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</p> <p><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 software per l'elaborazione dei segnali in tempo reale.</p> <p><i>Abilità comunicative</i> 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> 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-</p>	<p>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</p> <p><i>Making judgments:</i> 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> 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</p>	
--	---	---	--

		tattili, e alla progettazione di software per la resa di suono e vibrazioni.	non-visual interface component. - <i>Learning skills:</i> 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.	
Complementi di basi di dati e GIS	INF/01	<p>Il corso ha lo scopo di far acquisire allo studente una conoscenza approfondita di tematiche avanzate di gestione dei dati, specialmente per quanto riguarda la progettazione fisica, la sicurezza e l'analisi dei dati. Il corso mira inoltre a fornire specifiche competenze tecniche inerenti alle basi di dati geografiche, dai modelli concettuali spaziali alle strutture d'indicizzazione spaziale. Ulteriore obiettivo è presentare le tendenze più recenti nell'ambito della tecnologia delle basi di dati. Alla fine del corso lo studente sarà in grado di valutare le prestazioni delle basi di dati e attuare misure di correzione, nonché di realizzare adeguatamente le opportune politiche di sicurezza e privacy dei dati; avrà inoltre assimilato i concetti e le metodologie fondamentali per la configurazione di sistemi per basi di dati distribuite, per lo sviluppo di basi di dati geografiche e per l'analisi di dati spaziali.</p> <p>Capacità relative alle discipline</p> <p>1.1. Conoscenza e capacità di comprensione</p> <ul style="list-style-type: none"> • Modelli concettuali e logici per i dati spaziali e geografici. • Metodi d'indicizzazione di dati spaziali. • Architetture per sistemi paralleli e sistemi distribuiti di basi di dati. 	<p>The overall aim of the course is to acquire an in-depth knowledge on advanced topics in data management, especially those related to physical design, security and data analysis. Another goal is to provide adequate technical competences about geographical database management systems, from spatial conceptual models to spatial indexing structures. Yet another goal is to gain a thorough understanding of state-of-the-art database technologies and recent trends. At the end of the course, the student will be able to evaluate and tune the performance of a database, will have built up the ability to correctly implement suitable security and privacy policies, and will have learned the concepts and methodologies for the configuration of distributed databases, the development of geographical databases, and for the analysis of spatial data.</p> <p>Sector-specific skills</p> <p>1.1. Knowledge and understanding</p> <ul style="list-style-type: none"> • Conceptual and logical models for spatial and geographical data. • Spatial indexing methods. 	

	<ul style="list-style-type: none"> • 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. • Flussi d'informazione, controllo degli accessi, controllo delle inferenze in basi di dati statistiche. • 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"> • Uso di sistemi GIS (QGIS). • Realizzazione di basi di dati geografiche. • SQL spaziale. • Analisi di dati spaziali, quali, ad esempio, risoluzione di problemi di <i>nearest neighbor</i> o problemi di <i>routing</i>. • Tecniche di ottimizzazione delle prestazioni di un sistema di gestione dei dati relazionale. • Implementazione di politiche di protezione dei dati. • Processamento dei dati con sistemi non relazionali (e.g., MapReduce). <p>Capacità trasversali / soft skills</p> <p>2.1. Autonomia di giudizio</p> <ul style="list-style-type: none"> • Analizzare e formalizzare i requisiti di un sistema informativo geografico. • Saper scegliere le tecniche corrette e gli strumenti appropriati per condurre analisi di dati spaziali. • Capacità di determinare l'architettura (centralizzata, parallela, distribuita, 	<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. • Information flows, access control, inference control in statistical databases. • Features of new generation (NoSQL, NewSQL) systems. <p>1.2. Applying knowledge and understanding</p> <ul style="list-style-type: none"> • Usage of GIS systems (QGIS). • Implementation of geographical databases. • Spatial SQL. • Spatial data analysis, e.g., nearest neighbor problems and routing problems. • Optimization techniques for performance improvement in relational systems. • Implementation of data protection policies. • Data processing in non-relational systems (e.g. MapReduce). <p>Cross-sectoral skills/soft skills</p> <p>2.1. Making judgments</p> <ul style="list-style-type: none"> • Analyze and formalize the requirements of a geographical information system. • Choose the correct techniques and the appropriate tools to carry out spatial data analyses. 	
--	---	--	--

		<p>relazionale o meno) più adatta a risolvere un problema di gestione dei dati.</p> <ul style="list-style-type: none"> • Porre in essere le strategie più opportune per migliorare le prestazioni delle interrogazioni. • Valutare potenziali debolezze nelle politiche e nei meccanismi per garantire la protezione dei dati. <p>2.2. Abilità comunicative</p> <ul style="list-style-type: none"> • Saper comunicare attraverso il lessico tecnico dei sistemi GIS. • Saper comunicare usando la terminologia dei sistemi paralleli e distribuiti. • Saper comunicare con i soggetti direttamente e indirettamente coinvolti nella sicurezza e privacy dei dati. <p>2.3. Capacità di apprendimento</p> <ul style="list-style-type: none"> • Saper leggere e comprendere articoli scientifici inerenti ai temi del corso. • Saper ottimizzare un sistema di gestione dei dati, anche parallelo o distribuito, ovvero non relazionale. 	<ul style="list-style-type: none"> • 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. • Evaluate potential weaknesses in the policies and mechanisms that should guarantee the protection of data. <p>2.2. Communication skills</p> <ul style="list-style-type: none"> • Communicate using the technical lexicon of GIS systems. • Communicate using the terminology of parallel and distributed systems. • Communicate with the (technical and non-technical) stakeholders involved in data security and privacy. <p>2.3. Learning skills</p> <ul style="list-style-type: none"> • Read and understand scientific papers about the topics discussed in the course. • Learn to optimize a (possibly parallel or distributed, and necessarily relational) data management system. 	
Complessità e teoria dell'informazione	INF/01	<p>Indice:</p> <ul style="list-style-type: none"> • Teoria della Complessità <ul style="list-style-type: none"> ○ <i>Complessità in termini di Tempo e Spazio su Macchine di Turing e altri modelli classici</i> 	<p>Index:</p> <ul style="list-style-type: none"> • Complexity Theory <ul style="list-style-type: none"> ○ <i>Time and Space complexity on Turing Machines and other classical models</i> 	

	<ul style="list-style-type: none"> ○ <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 in tempo e spazio. Elaborare riduzioni tra i linguaggi. Definire e implementare algoritmi sui grafi per varianti dei problemi analizzati nel</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 classical models. Describe algorithms on graphs.</p>	
--	---	---	--

		<p>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>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</p>	
--	--	--	--	--

			<p>related problems. Exploit new instruments for improving the computational complexities. Identify information, coding and communication problems/solutions.</p>	
<p>Geometria Computazionale</p>	<p>INF/01</p>	<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. Particolare attenzione è rivolta all'analisi della correttezza e della complessità computazionale degli algoritmi discussi. 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. Programma Sintetico Problemi di robustezza dell'elaborazione di dati geometrici. Approcci generali alla soluzione di problemi di geometria piana: - Approccio divide-et-impera; - Tecniche di plane sweep; - Tecniche incrementali randomizzate. Problemi notevoli nell'ambito della geometria computazionale piana e analisi dei costi computazionali: - Convex hull; - Intersezioni di segmenti; - Partizioni di regioni poligonali: triangolazioni, mappe trapezoidali e suddivisioni monotone;</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. The main approaches being introduced exploit divide-et-impera, plane-sweep and randomized-incremental techniques. Particular attention is given to the analysis of the algorithms' correctness and computational costs. 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. Short Syllabus Robustness issues of geometric data processing. General approaches to planar geometry problems: - Divide-et-impera approach; - Plane-sweep techniques; - Randomized-incremental techniques. Noteworthy problems of planar computational geometry and analysis of the related computational costs: - Convex hull; - Segment intersection;</p>	

	<ul style="list-style-type: none"> - Problemi di point-location; - Problemi di prossimità: diagrammi di Voronoi e triangolazioni di Delaunay. <p>Capacità relative alle discipline Lo/la studente/essa dovrà:</p> <p>Capacità relative alla disciplina</p> <p><i>1.1. 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; - 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. 	<ul style="list-style-type: none"> - 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><i>1.1. Knowledge and understanding</i></p> <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>	
--	--	--	--

		<p><i>2.2 Abilità comunicative.</i> - Essere in grado di usare un linguaggio preciso per presentare i risultati dell'analisi delle proprietà di un algoritmo.</p> <p><i>2.3 Capacità di apprendimento</i> - Essere in grado di orientarsi nell'ambito della geometria computazionale e di studiare autonomamente nuovi algoritmi.</p>	<p>- Being able to use an accurate language to present the results of the analysis of an algorithm's properties.</p> <p><i>2.3 Learning skills</i> - Being able to study autonomously new computational geometry algorithms.</p>	
Immagini e multimedialità	ING-INF/05	<p>Obiettivi formativi specifici</p> <p><i>Immagini e loro rappresentazioni</i> Immagini statiche 2D: rappresentazioni numeriche, statistiche, grafiche. L'ambiente di elaborazione software MATLAB (©The Mathworks).</p> <p><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)</p> <p><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:</p> <p>Capacità relative alle discipline</p> <p><i>1.1 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</p>	<p>Teaching goals</p> <p><i>Images and their representations</i> Static 2D images: numerical, statistical, graphical representations. The MATLAB (©The Mathworks) software environment.</p> <p><i>Image Processing Algorithms</i> Point operators: image transformations. Neighborhood operators: image digital filtering. Image enhancement techniques Region segmentation and feature extraction.</p> <p><i>Graphic 3D Modelling</i> The JAVA3D™ modelling environment Geometric transforms Geometric object modelling Lighting and shadowing Interactions; animations. The student is expected to:</p> <p>Sector-specific skills</p> <p><i>1.1 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</p>	

	<p>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.</p> <p><i>1.2 Capacità di applicare conoscenza e comprensione.</i> Saper lavorare in un ambiente di 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 (scenograph). Saper realizzare il modello tramite un linguaggio a oggetti.</p> <p>Capacità trasversali / soft skills</p> <p><i>2.1 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><i>2.2 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</p>	<p>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.</p> <p><i>1.2 Applying knowledge and understanding.</i> Be able to work into an image processing software environment. Be able to select and use numerical-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><i>2.1 Making judgements.</i> Be able to critically analyze an image processing technique. Be able to select the appropriate software tools and libraries.</p> <p><i>2.2 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><i>2.3 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>	
--	--	--	--

		ambito di standard tecnologici e ambienti di sviluppo.		
Informatica e aziende	INF/01	<p>Informatica e Aziende propone uno sguardo sull'ambiente in cui i sistemi informativi si inseriscono (il DOVE – l'organizzazione in cui l'oggetto del lavoro si colloca). Assieme a "Sistemi Informativi e Data Warehouse" e "Ingegneria del software – progettazione e laboratorio II" compone un 'cluster' di corsi dai contenuti coerenti intorno ai temi dell'Informatica in Azienda. Il corso si dispiega attraverso lezioni frontali, analisi di casi, letture specifiche e interventi di professionisti con una visione strategica sui propri sistemi informativi. La trattazione riguarda sia gli aspetti concettuali, sia le applicazioni pratiche. Il corso è diviso nelle seguenti porzioni</p> <p><i>1. Le componenti strutturali</i></p> <p>Sono g elementi immediatamente 'riconoscibili' quando si parla di Informatica in Azienda</p> <ul style="list-style-type: none"> - La struttura delle aziende: dimensioni, mercato, articolazione topologica e organizzativa - Natura e dinamica dei flussi informativi in azienda. Influenza della tecnologia sulla modellazione dell'organizzazione aziendale. - Cenni sui sistemi informativi aziendali: ERP e i suoi satelliti (Qualità, Finanza, Risorse, ...); I sistemi di supporto alle decisioni; Big Data in azienda; gli strumenti di supporto alle core business, alle attività di office, alle comunicazioni; i generatori di dati (controllo di produzione, IoT, web) 	<p>Specific training targets</p> <p>The course "ICT and Business Companies" offers an insight on the environment in which information systems fit (the WHERE).</p> <p>Together with "Information Systems and Data Warehouse" and "Software Engineering - Design and Laboratory II", it makes a 'cluster' of courses that develop interconnected themes on the application of IT issues in Business Companies. The course flows through frontal lessons, case studies, specific readings and seminars with IT people that give their strategic view on information systems. The lessons cover both conceptual aspects and practical applications. The course is composed by the following sections</p> <p><i>1. Structural components</i></p> <p>Talking of ICT in Business Companies, structural components are the most visible themes:</p> <ul style="list-style-type: none"> - Organization structure: dimensions, market, topology and layout, official roles - Nature and dynamics of information flows in the company. Influence of technology on corporate organization modelling. - Corporate information systems: ERP and its satellites (Quality, Finance, Resources, ...); Decision support systems; Big Data in the 	

	<ul style="list-style-type: none"> - L'infrastruttura tecnologica (di produzione, di comunicazione, di servizio, di protezione/controllo) – componenti, topologie. Punti critici rispetto alla realtà aziendale. Virtualizzazione/scalabilità/ridondanza. Soluzioni in Cloud <p><i>2. I temi etici</i> Le questioni etiche sollevate dall'uso di ICT in azienda, tra elementi normati e sensibilità culturali</p> <ul style="list-style-type: none"> - Sicurezza, Riservatezza, Responsabilità, Proprietà ed i relativi strumenti normativi e di indirizzo - Cenni sugli aspetti peculiari dell'ICT che originano nuovi dilemmi etici. <p><i>2. I temi relazionali</i> L'azienda come luogo di relazione, sostenuto dalla tecnologia</p> <ul style="list-style-type: none"> - L'ecosistema azienda: la struttura informale dell'organizzazione. - Le persone tra obiettivi propri e obiettivi aziendali. - Relazioni all'interno e con l'esterno. Culture personali e organizzative. - Contesto interno, sociale e di mercato. - Propensione aziendale al cambiamento <p><i>3. I temi del cambiamento</i> Informatica come veicolo di trasformazione dell'azienda. Il cambiamento tra necessità e opportunità.</p> <ul style="list-style-type: none"> - Innovazione (di processi e di tecnologia) indotta dal contesto politico/sociale, dal contesto di mercato, suggerita dall'evoluzione tecnologica 	<p>company; Core business support tools, Office support tools, Communications; Data generators (production control, IoT, Web)</p> <ul style="list-style-type: none"> - Technological infrastructure (for production, communication, service, protection / control): components and topologies. Critical issues related to the business contest. Virtualization / scalability / redundancy. Cloud solutions <p><i>2. ICT and ethics</i> Ethical issues raised by the use of ICT in business companies, among normative elements and cultural propensities</p> <ul style="list-style-type: none"> - Security, Confidentiality, Responsibility, Property. Laws, social rules and policies - Peculiar aspects of ICT rising new ethical dilemmas. <p><i>3. Relations</i> Business Company as a place of relationship, sustained by technology</p> <ul style="list-style-type: none"> - Company's ecosystem: the informal structure of the organization - People between their own goals and business goals - Relations inside and out. Personal and organizational cultures. - Internal, social and market context. - Business propensity to change <p><i>4. Innovation and change</i></p>	
--	--	--	--

		<ul style="list-style-type: none"> - Pilotare i processi di cambiamento IT: relazione tra l'incarico degli interventi e l'organizzazione, tra le persone e le richieste di cambiamento. Cenni su pianificazione e conduzione del progetto. I rischi e le resistenze al cambiamento - I tempi del cambiamento IT – dalle svolte epocali alla trasformazione continua. Cenni sulle innovazioni del prossimo futuro e sulle potenziali ricadute/trasformazioni a livello aziendale <p>Capacità relative alle discipline</p> <p>1.1 <i>Conoscenza e capacità di comprensione</i>: lo studente acquisisce conoscenze specifiche sulle organizzazioni, sul ruolo e l'impatto delle tecnologie ICT all'interno dei processi di lavoro e delle relazioni aziendali</p> <p>1.2 <i>Conoscenza e capacità di comprensione applicate</i>: lo studente sviluppa capacità di lettura di contesti sociali, quali sono le aziende, e delle dinamiche sottese da processi di cambiamento indotti dall'introduzione di ICT</p> <p>Capacità trasversali/soft skills</p> <p>2.1 <i>Autonomia di giudizio</i>: lo studente acquisisce una capacità di valutazione rispetto alle tecnologie e agli approcci di innovazione applicabili in relazione a diverse culture organizzative</p> <p>2.2 <i>Abilità comunicative</i>: lo studente impara la terminologia propria legata alle organizzazioni aziendali e la impiega correttamente nell'ambito dell'uso di soluzioni ICT in azienda.</p>	<p>ICT as a company transformation vector. The change between need and opportunity.</p> <ul style="list-style-type: none"> - Innovation driven by the political / social context, by the market context, suggested by technological evolution - Driving IT change processes: relationship between people in charge of change process and the organization, between people and requests for change. Planning and project management. Risks and resistance to change - The times of IT change - from epoch-making to continuous transformation. Insights into the future innovations and their potential impacts / transformations on business companies <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding</i>: the student acquires specific knowledge on organizations, the role and impact of ICT technologies on work processes and business relationships</p> <p>1.2 <i>Applying knowledge and understanding</i>: The student develops the ability to read social contexts, such as companies, and the dynamics underlying the processes of change induced by ICT</p> <p>Cross-sectoral/soft skills</p> <p>2.1 <i>Making judgments</i>: the student develops evaluation skills on technologies</p>	
--	--	--	--	--

		<p>2.3 <i>Capacità di apprendimento</i>: lo studente estende la sua autonomia di apprendimento e le sue capacità di lettura trasversale dei contesti attraverso l'analisi dei casi, la lettura di testi, il confronto nel lavoro di gruppo</p>	<p>and innovation approaches applicable to different organizational cultures</p> <p>2.2 <i>Communication skills</i>: the student develops a specific vocabulary related to business companies and is able to use it correctly in the scope of information systems.</p> <p>2.3 <i>Learning Skills</i>: The student extends his autonomy of learning and his / her cross-reading skills through case analysis, reading of texts, work groups.</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>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</p> <p>Conoscere le caratteristiche di base del dominio giuridico, sia dal punto di vista della</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 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 	

		<p>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</p> <p>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</p> <p>Lo/la studente/ssa dovrà:</p> <p>2.1 Autonomia di giudizio</p> <p>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</p> <p>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</p> <p>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</p>	<ul style="list-style-type: none"> ● 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:</p> <p>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>To be able to evaluate and design IT applications in the fields of e-government, e-legislation and e-democracy.</p> <p>To be able to identify and use standards and technologies suitable for a given legal context.</p> <p>Cross-sectoral skills/soft skills</p> <p>At the end of the course the student is expected to:</p> <p>2.1 Making judgements:</p> <p>Acquire the capacity to assess technologies and innovation approaches appropriate to the legal environment.</p> <p>2.2 Communication skills:</p> <p>Know the legal terms to be able to interact properly and understand the needs</p>	
--	--	---	--	--

		informarsi autonomamente consultando la letteratura scientifica.	underlying the requirements of a computer system applied in a legal context. 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.	
Informatica medica	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.</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</p>	

	<p>Conoscere funzioni e criticità dei sistemi di eHealth e Telemedicina.</p> <p>1.2 Capacità di applicare conoscenza e comprensione</p> <p>Conoscere le caratteristiche di base del sistema sanitario nazionale.</p> <p>Saper identificare ed utilizzare gli standard e le tecnologie applicabili in un determinato contesto biomedico.</p> <p>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.</p> <p>Saper impostare un processo di elaborazione ed analisi di bioimmagini.</p> <p>Saper definire un sistema di telemedicina appropriato secondo le Linee Guida nazionali.</p> <p>Capacità trasversali / soft skills</p> <p>Lo/la studente/ssa dovrà:</p> <p>2.1 Autonomia di giudizio</p> <p>Saper esaminare ed inquadrare correttamente un determinato problema biomedico.</p> <p>2.2 Abilità comunicative.</p> <p>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.</p> <p>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</p> <p>Saper reperire ed utilizzare risorse informatiche</p>	<p>biomedical knowledge, as well as of the organization of healthcare services and their informatization.</p> <p>know functions and issues of eHealth and telemedicine systems.</p> <p>1.2 Applying knowledge and understanding</p> <p>To know basic features of the Italian Health care systems, as example of organization.</p> <p>To be able to identify and use standards and technologies suitable in a specific biomedical context.</p> <p>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.</p> <p>To be able to design a biomedical image processing and analysis system.</p> <p>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</p> <p>Be able to study and correctly frame a specific biomedical problem.</p> <p>2.2 Communication skills.</p> <p>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.</p> <p>Be able to motivate the conceptual choices made when designing an informatic solution to a biomedical problem.</p>	
--	---	---	--

		<p>e scientifiche per l'approfondimento autonomo delle tematiche studiate a lezione. Saper leggere ed approfondire un articolo scientifico in ambito biomedico.</p>	<p>2.3 Learning skills Be able to find and use informatic and scientific resources for an autonomous close examination of the course topics. Be able to read and delve into a biomedical scientific paper.</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> <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); 	<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> <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"> ● 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 <i>Conoscenza 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 di base sia le correnti linee di ricerca e le tendenze future della disciplina.</p> <p>1.2 <i>Conoscenza e capacità di comprensione applicate:</i> 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 <i>Autonomia di giudizio:</i> 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 <i>Abilità comunicative:</i> 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</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 <i>Knowledge and understanding:</i> the student will acquire the basic knowledge of the IR field and of related topics. The student will know both basic topics and advanced research trends of the field, with some hints at future research developments.</p> <p>1.2 <i>applying knowledge and understanding:</i> 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 <i>Making judgements:</i> the student acquires the skills to critically evaluate how technologies, algorithms, and</p>	
--	--	---	--

		<p>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>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 is also able to autonomously obtain information from the scientific literature.</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 Conoscenza e comprensione: l'insegnamento si propone di fornire le conoscenze</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>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 	
Interactive 3D Graphics	ING-INF/05	Dai dispositivi mobili fino ai sistemi per la Realtà Virtuale, sempre più applicazioni si basano sulla capacità di visualizzare, in modo realistico,	From mobile devices to virtual reality systems, more and more applications are based on the capability to visualize 3D	

		<p>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>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 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 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>	
Modelli e algoritmi per le decisioni	INF/01	<p>L'obiettivo culturale del corso è quello di presentare le principali metodologie modellistiche utilizzate nella risoluzione di problemi computazionalmente difficili che si presentano in svariati contesti del mondo reale (quali, ad es., produzione, allocazione, pianificazione, schedulazione). In particolare, si analizzeranno tre linee di attacco a tali problemi: (i) algoritmi esatti, (ii) algoritmi approssimati e (iii) algoritmi euristici (ricerca locale). Relativamente agli algoritmi esatti verrà data particolare enfasi alla tecnica del branch-and-bound. Gli algoritmi approssimati verranno descritti tramite numerosi esempi su alcuni classici</p>	<p>The goal of the course is to describe the main modeling techniques used for the solution of hard computational problems from real life (such as production, planning, allocation, scheduling). In particular, we will address three lines of solution for such problems: (i) exact algorithms, (ii) approximation algorithms and (iii) heuristics (local search). With respect to exact algorithms, we will put particular emphasis on branch-and-bound. Approximation algorithms will be described via several examples of classical optimization problems. Finally, we will describe some among the most successful</p>	

	<p>problemi di ottimizzazione. Infine, verranno esposte alcune fra le meta-euristiche di maggior successo, quali la tabu-search e gli algoritmi genetici. Al termine del corso, lo studente dovrebbe essere in grado di descrivere un modello matematico per un generico problema di ottimizzazione combinatoria e proporre adeguati strumenti di risoluzione dello stesso.</p> <p>Capacità relative alle discipline:</p> <p>1.1 <i>Conoscenza e capacità di comprensione:</i> lo studente acquisisce la conoscenza degli approcci algoritmici efficaci e di eventuali strumenti software disponibili online per la risoluzione di problemi di ottimizzazione combinatoria. Impara inoltre a valutare la difficoltà di un problema e a scegliere la strategia più adatta alla sua risoluzione.</p> <p>1.2 <i>Conoscenza e capacità di comprensione applicate:</i> il corso fornisce numerosi esempi per portare lo studente ad essere in grado di applicare a casi reali le conoscenze maturate sugli esempi didattici.</p> <p>Capacità trasversali/soft skills</p> <p>2.1 <i>Autonomia di giudizio:</i> lo studente acquisisce una capacità di valutazione critica sulla difficoltà dei problemi e sugli approcci da utilizzarsi, di volta in volta, per la loro soluzione.</p> <p>2.2 <i>Abilità comunicative:</i> lo studente impara a descrivere in modo rigoroso e matematicamente corretto un problema e i suoi vari dettagli, e ad usare la terminologia appropriata per lo studio e la realizzazione di una soluzione algoritmica.</p> <p>2.3 <i>Capacità di apprendimento:</i> lo studente impara ad essere autonomo nell'espandere le proprie conoscenze sui concetti di complessità e</p>	<p>meta-heuristic techniques, such as taboo search and genetic algorithms. By the end of the course, the student should be able to describe a mathematical model for a generic combinatorial optimization problem and to propose suitable tools for its solution.</p> <p>Sector-specific skills</p> <p>1.1 <i>Knowledge and understanding</i> The student acquires knowledge of the effective algorithmic approaches and of some potential software tools for the solution of combinatorial optimization problems. Furthermore, he learns how to access the difficulty of a problem and to choose the most appropriate strategy for its solution.</p> <p>1.2 <i>Applying knowledge and understanding</i> The course gives several examples that lead the student to become able of applying the skills acquired on the examples to real-life cases.</p> <p>Cross-sectoral skills/soft skills</p> <p>2.1 <i>Making judgements</i> The student acquires a critical ability to access the difficulty of a problem and to point to an appropriate approach for its solution.</p> <p>2.2 <i>Communication skills</i> The student learns how to describe a problem and its details in a rigorous, mathematically sound way, and how to use the correct terminology for the study</p>	
--	--	--	--

		<p>di analisi dei problemi e relativi algoritmi. Nel fare ciò sa anche attingere a nozioni acquisite in altri corsi focalizzati su algoritmi e strutture dati efficienti e/o sulla programmazione.</p>	<p>and implementation of an algorithmic solution.</p> <p><i>2.3 Learning skills</i></p> <p>The student learns how to become autonomous in expanding his knowledge about the complexity and analysis of problems and algorithms. He also utilizes notions from other courses on algorithms and effective data structures and programming.</p>	
<p>Progettazione di applicazioni mobili</p>	<p>INF/01</p>	<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 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 la piattaforma Android.</p> <p>DISPOSITIVI E PIATTAFORME MOBILI</p> <p>Tipi di dispositivi mobili. Smartphone, feature phone, tablet e app: mercato, differenze, dati statistici. Caratteristiche e limitazioni dei dispositivi mobili: CPU, GPU, memoria, display, input, alimentazione, connettività. Principali piattaforme mobili e loro caratteristiche: Apple iOS, Google Android, Windows Phone.</p> <p>Applicazioni mobili native vs. web-based vs. ibride: caratteristiche e tecnologie utilizzabili.</p> <p>PROGETTAZIONE DI APPLICAZIONI MOBILI</p> <p>Ciclo di sviluppo di un sistema interattivo. Design dell'interazione, user experience (usability,</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 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 Android platform.</p> <p>MOBILE DEVICES AND PLATFORMS</p> <p>Overview of the different types of mobile device. Smartphones, feature phones, tablets and apps: market, peculiarities, statistical data. Characteristics and limitations of mobile devices: CPU, GPU, display, memory, input, energy supply, connectivity. Major mobile platforms and their features: Apple iOS, Google Android, Windows Phone. Native vs. web-based vs. hybrid mobile applications: features and technologies.</p>	

		<p>usefulness, emotional impact), the UX lifecycle process, UX roles. Il diagramma della complessità, complessità di dominio e dell'interazione, sistemi a bassa complessità. Approccio leggero alla progettazione di applicazioni mobili: il ciclo envision, prototype, evaluate. Envision tramite storyboarding: processo e linee guida per la creazione di uno storyboard, esempi. Personas e altre tecniche complementari: definizione, creazione, utilizzo, esempi. Low-fidelity prototyping tramite wireframing: descrizione, esempi. Valutazione di prototipi low-fidelity: scelta degli utenti, design alternativi, processo di valutazione. Hi-fidelity prototyping: definizione, esempi. Design pattern per la progettazione di applicazioni e servizi mobili: funzionalità di navigazione, tutorial, anti-pattern comuni, altri pattern.</p> <p>SVILUPPO DI APPLICAZIONI MOBILI</p> <p>Sviluppo di applicazioni Android: strumenti di sviluppo (Android Studio, Android SDK, emulatore), ciclo di vita delle applicazioni Android, componenti di un'applicazione Android (activity, service, content provider, broadcast receiver), interfacce utente (layout XML, widget di base, risorse, menu, elementi d'interfaccia avanzati), intent, meccanismi di data storage e retrieval, servizi di localizzazione, multithreading.</p> <p>Al termine del corso, lo/la studente/essa dovrà:</p> <ul style="list-style-type: none"> - conoscere le tecnologie disponibili per la realizzazione di applicazioni e servizi per dispositivi mobili 	<p>MOBILE APPLICATION DESIGN</p> <p>Development cycle of an interactive system. Designing the interaction, user experience (usability, usefulness, emotional impact), the UX lifecycle process, UX roles. The complexity diagram, domain and interaction complexity, low-complexity systems. A lightweight approach to the design of mobile applications: the envision, prototype, evaluate cycle. Envision through storyboarding: process and guidelines for creating a storyboard, examples. Personas and other complementary techniques: definition, creation, use, examples. Low-fidelity prototyping through wireframing: description, examples. Evaluation of low-fidelity prototypes: user choice, design alternatives, evaluation process. Hi-fidelity prototyping: definition, examples. Design patterns for mobile applications and services: navigation, tutorials, common anti-patterns, other patterns.</p> <p>MOBILE APPLICATION DEVELOPMENT</p> <p>Android application development: development tools (Android Studio, Android SDK, emulator), life cycle of Android applications, components of an Android application (activity, service, content provider, broadcast receiver), user interfaces (XML layouts, base widgets, resources, menus, advanced interface elements), intents, data storage and retrieval, location services, multi-threading.</p>	
--	--	--	--	--

	<ul style="list-style-type: none"> - conoscere le fasi fondamentali della progettazione dell'interazione con applicazioni mobili - conoscere le soluzioni d'interfaccia più utilizzate per l'interazione con applicazioni e servizi per dispositivi mobili - conoscere i dettagli implementativi fondamentali di applicazioni e servizi mobili per la piattaforma Android - saper identificare le tecnologie più appropriate per lo sviluppo di applicazioni e servizi mobili nell'attuale contesto di mercato - saper progettare l'interazione con applicazioni e servizi mobili sfruttando appropriate soluzioni d'interfaccia - saper implementare applicazioni e servizi mobili usabili per la piattaforma Android <p>Capacità relative alle discipline:</p> <ul style="list-style-type: none"> - <i>Conoscenza e comprensione</i> <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 su piattaforma Android.</p>	<p>At the end of the course, students will:</p> <ul style="list-style-type: none"> - know the available technologies for the development of applications and services for mobile devices - know the key phases in the design of interaction with mobile applications - know the most common interface solutions for interaction with applications and services for mobile devices - know the core implementation details of Android applications and services - be able to identify the most appropriate technologies for the development of applications and services for mobile devices - be able to design the interaction with applications and services for mobile devices, using the most appropriate interface solutions - be able to develop usable applications and services for the Android platform <p>Sector-specific skills:</p> <ul style="list-style-type: none"> - <i>Knowledge and understanding</i> <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</p>	
--	---	---	--

	<p>- <i>Capacità di applicare conoscenza e comprensione</i> 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 alla specifica applicazione/servizio mobile considerati, evitando l'utilizzo di comuni anti-pattern; capacità di sviluppare applicazioni/servizi mobili interattivi per la piattaforma Android.</p> <p>Capacità trasversali/soft skills:</p> <p>- <i>Autonomia di giudizio</i> 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.</p> <p>- <i>Abilità comunicative</i> 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</p>	<p>implementation details for the development of Android applications and services.</p> <p>- <i>Applying knowledge and understanding</i> 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 interface patterns for the considered mobile application/service, avoiding common anti-patterns; the ability to develop interactive mobile applications/services for the Android platform.</p> <p>Cross-sectoral skills/soft skills:</p> <p>- <i>Making judgements</i> 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.</p> <p>- <i>Communication skills</i></p>	
--	--	---	--

		<p>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.</p> <p>- <i>Capacità di apprendimento</i> 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>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.</p> <p>- <i>Learning skills</i> 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>	
Progettazione di sistemi multimediali	ING-INF/05	<p>Il corso ha tre obiettivi fondamentali: (i) approfondire l'aspetto strutturale e funzionale degli oggetti multimodali sia dal punto di vista teorico sia tecnico; (ii) fornire criteri e metodi per affrontare il problema della selezione e della integrazione dei media; (iii) presentare alcune metodologie recenti per la progettazione di multimedia e transmedia orientate alla esperienza dell'utente (Design for Experience) e all'adattamento culturale (Cross-cultural Design).</p> <p>L'approccio seguito è di tipo semiotico e si focalizza sulla costruzione, comunicazione e interpretazione di significati attraverso la composizione nello spazio e nel tempo di testi, immagini, suoni, video e audiovisivi.</p> <p>Alla fine del corso lo/la studente/essa dovrà:</p> <p>Capacità relative alle discipline <i>Conoscenza e capacità di comprensione</i> Conoscere le concettualizzazioni di base e il vocabolario tecnico per l'analisi e la progettazione di prodotti multimediali</p>	<p>The course has three main objectives: (i) to study the structural and functional features of multimodal artifacts from a theoretical and technical point of view; (ii) to provide criteria and methods for addressing the problem of selection and integration of media; (iii) to present and critically discuss some recent approaches and methodologies for multimedia and transmedia design proposed within the fields of "Design for Experience" and "Cross-cultural Design". The methodological approach is inspired to semiotic theories and focuses on meaning construction, communication and interpretation of multimodal resources</p> <p>At the end of the course, the student should be able</p> <p>Sector-specific skills <i>Knowledge and understanding</i></p>	

		<p>Conoscere i principali approcci, metodi e strumenti per affrontare un problema di progettazione nel campo multimediale (e transmediale) finalizzato a specifici obiettivi e intenti (es. progettazione per l'esperienza, progettazione cross-culturale, progettazione per i valori)</p> <p><i>Conoscenza e capacità di comprensione applicate</i></p> <p>Saper applicare le conoscenze concettuali e metodologiche di cui sopra a situazioni di progettazione reali relative a diversi generi di prodotti multimediali (es. siti web, video commerciali, giochi) e a diverse tipologie di utenti e contesti d'uso</p> <p>Capacità trasversali/soft skills</p> <p><i>Autonomia di giudizio</i></p> <p>Analizzare, confrontare e valutare in maniera critica soluzioni di progetto da diversi punti di vista (strutturale, comportamentale, funzionale, applicativo, e sociale-culturale), secondo diversi criteri (usabilità, efficacia comunicativa, efficienza tecnica, conseguenze etiche e sociali) e a diversi livelli di aggregazione (intero prodotto, parti del prodotto rispetto al tutto, prodotto rispetto al contesto d'uso)</p> <p><i>Abilità comunicative</i></p> <p>Spiegare come un dato prodotto multimodale soddisfa gli scopi comunicativi per cui è stato realizzato</p> <p>Giustificare le scelte tecniche e di progetto</p> <p><i>Capacità di apprendimento</i></p> <p>Generalizzare/specializzare concetti, modelli e metodi</p>	<p>To know the conceptualizations and the technical vocabulary for the analysis and design of multimodal artifacts</p> <p>To know the main approaches, methodologies and tools for addressing a design problem in the multimedia and transmedia domains in order to achieve specific objectives and intents (e.g., design for human experience, design for culture, design for values and subjective wellbeing)</p> <p><i>Applying knowledge and understanding</i></p> <p>To apply conceptual and methodological knowledge to concrete design problems concerning different genres of multimodal artifacts (e.g., web sites and portals, video commercials, virtual exhibitions, games) and different kinds of users and contexts of use</p> <p>Cross-sectoral skills/soft skills</p> <p><i>Making judgments</i></p> <p>To compare, evaluate and criticize two or more design solutions from different points of view (i.e., structural, behavioral, functional, socio-cultural) according to different criteria (e.g., usability, communicative effectiveness, social and ethical effects) and at different aggregation and abstraction levels (e.g., the whole product, single parts with respect to the whole, the artifact with respect to the context of use)</p> <p><i>Communication skills</i></p> <p>To describe a design solution</p> <p>To explain how it achieves the intended goals and requirements</p>	
--	--	--	---	--

		<p>Stabilire analogie tra concettualizzazioni e trasferire concetti e metodi tra domini differenti in maniera opportunistica</p> <p>Usare il ragionamento abduttivo per formulare ipotesi di soluzioni a partire dalle specifiche di progetto</p> <p>Riflettere sul proprio percorso di progettazione (meta-riflessione)</p>	<p>To motivate design choices</p> <p><i>Learning skills</i></p> <p>To generalize/specialize concepts, models and methods</p> <p>To abductively propose design solutions starting from problem specifications</p> <p>To use analogies in order to transfer and apply design knowledge across application domains</p> <p>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). ● I principi della progettazione OO. I concetti e le tecniche che consentono di costruire progetti OO di buona qualità, ad esempio: domini, 	<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, component, and deployment diagrams. ● OO Design principles. Concepts and techniques leading to good OO design, for instance: 	

		<p>ingombro, coesione, spazio degli stati, comportamento, conformità di tipo, progetto per contratti, ecc.</p> <ul style="list-style-type: none"> ● 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</p> <p>1.1 <i>Conoscenza e capacità di comprensione:</i> lo studente acquisisce le conoscenze dei concetti e delle tecniche avanzati di analisi e progetto del mondo OO.</p> <p>1.2 <i>Conoscenza e capacità di comprensione applicate:</i> lo studente saprà progettare software</p>	<p>domains, encumbrance, cohesion, state space, behavior, type conformance, design by contract, etc.</p> <ul style="list-style-type: none"> ● 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>1.1 <i>Knowledge and understanding:</i> the student will know the advanced concepts and techniques for Object Oriented (OO) analysis and design.</p>	
--	--	---	--	--

		<p>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>1.2 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>	
Ragionamento automatico	INF/01	<p>Il corso affronta lo studio del ragionamento automatico nell'ambito dell'Intelligenza 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</p>	<p>Automated Reasoning will be introduced according to IEEE/ACM classification as a subfield of "Intelligent Systems". 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</p>	

	<p>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. 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>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. Relationships between expressivity and complexity. Action Description Languages and Planning.</p> <p>Sector-specific skills 1.1 Knowledge and understanding: the student will improve her/his capability of</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 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</p>	<p>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 specified goal, the student will know how to model and solve it.</p> <p>2.2 <i>Communication skills:</i> 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</p>	
--	--	--	--

		<p>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>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. - saper scegliere le tecniche di personalizzazione (adattativa) più adeguate. - saper analizzare un problema di accesso all'informazione 	<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 - Knowing how to select the most adequate personalization techniques - Knowing how to analyze a personalized information 	

		<p>personalizzato e proporre una possibile soluzione.</p> <p>Lo studente dovrà:</p> <p>Capacità relative alle discipline</p> <ol style="list-style-type: none"> 1. <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 2. <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>Capacità trasversali / soft skills</p> <p>Lo studente dovrà:</p> <ol style="list-style-type: none"> 1. <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 personalizzazione adattativa e valutarne la validità. 2. <i>Abilità comunicative:</i> saper illustrare con rigore logico e terminologico, a voce e per iscritto, questioni tecniche inerenti tecniche 	<p>access problem and how to propose a solution</p> <p>The student will have to:</p> <p>Sector-specific skills</p> <ol style="list-style-type: none"> 1. <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. 2. <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>Cross-sectoral skills/soft skills</p> <p>The student will have to:</p> <ol style="list-style-type: none"> 1. <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. 2. <i>Communication skills:</i> acquiring the ability to describe effectively and through appropriate models the scenario of an adaptive personalized system, its benefits and risks 	
--	--	---	--	--

		<p>e sistemi di personalizzazione adattativa dei contenuti Web.</p> <p>3. <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>3. <i>Learning skills</i>: being able to learn the overall functioning of recommender systems and their implications.</p>	
Robotica	ING-INF/05	<p>Lo scopo del corso è quello di fornire allo studente strumenti pratici per comprendere e implementare programmi che siano in grado di controllare veicoli autonomi. Viene presentata la nozione generale di agente situato introducendo concetti come sistemi basati sul comportamento, l'accoppiamento diretto della percezione all'azione, l'interazione dinamica con l'ambiente. Il focus dell'attenzione è particolarmente orientato al controllo dei veicoli autonomi discutendo un certo numero di esempi presenti nella letteratura tecnica. Viene anche presentato il problema della locomozione per i cosiddetti robot articolati discutendo di vari tipi di strutture meccaniche, inclusi gli umanoidi.</p>	<p>The aim of the course is that to provide a student with practical tools to understand and implement programs, which are able to control autonomous vehicles. The general notion of situated agent is presented by introducing concepts like behavior-based systems, the direct coupling of perception to action, the dynamical interaction with the environment. The focus of attention is especially oriented to autonomous vehicle control by discussing a number of distinguishing examples from the technical literature. The locomotion problem for the so-called legged-robots is also presented discussing various kinds of mechanical structures, humanoids included.</p> <p>2. Method</p> <p>The course is intended to understand the underlying principles, which distinguish robot control programs from computer programs. Their specific features are introduced starting from the point of view of the behavior-based systems starting from the direct coupling of perception to action and the dynamic interaction with the environment. In this perspective, a</p>	

			<p>general method to implement autonomous robot control is discussed, including legged and humanoid robotics. Multirobot systems are also presented. A number of examples from autonomous robotics are discussed including details of vehicle control with some possible experimental lectures using available moving platforms. Students are encouraged to gain insights starting from the topics presented during lectures and to make guided experimentations.</p> <p>3. Contents</p> <p>The topics are primarily based on situated agents, behavior-based architectures, reactive systems but the connection with A.I. and planning are also discussed. Biological motivations, adaptive and collective behaviors are introduced but also industrial manipulators performing an automatic assembly are presented and discussed. A preliminary In the following you can find a preliminary program of the course. A more precise scheduling of the lectures will appear before the beginning of the first one.</p> <p>Situated Agents</p> <ul style="list-style-type: none"> Sensing and Acting Planning Reactive Systems Biological Motivations Environment Constraints through Agent Embodiment <p>Autonomous Vehicles</p> <ul style="list-style-type: none"> Behavior-based Architectures 	
--	--	--	--	--

			<p>Mobile Robotics Control Issues Applications</p> <p>Dynamical Agents Dynamical System Review Mobile Vehicles Representational Issues Energy-based Methods for Movement Controls</p> <p>Implementing Behaviors Perceptual Basis for Behavior-based Control Representational Issues Reactive Architectures Hybrid Architectures Adaptive Behaviors Collective Behaviors</p> <p>Locomotion Wheel-driven vehicles Legged robots Exapodes, four-legged and bipedal robots Humanoid: a brief introduction Active and Passive Walking Gait control using the inverted pendulum model</p>	
Scienza delle reti	INF/01	<p>Le reti sono pervasive nel mondo reale (e anche immaginario). La natura, la società, l'informazione, la tecnologia, e talvolta persino l'arte sono supportate da diverse reti che condividono tra loro molte proprietà strutturali. Nel corso studieremo e applicheremo tecniche di analisi e di visualizzazione di reti. Useremo principalmente il pacchetto igraph dell'ambiente R e il linguaggio Processing. Gli argomenti trattati saranno i seguenti: * The origins of networks and networks in the contemporary world</p>	<p>Networks are pervasive in the real (and imaginary) world. Nature, society, information, technology, and sometimes even the arts are supported by different networks that share each other many structural properties. During the course we will investigate and apply analytical and visualization techniques for networks. We will learn the R environment (in particular the igraph package) and the language Processing. The contents of the course is the following:</p>	

		<p>* Introduction to Graph Theory and Linear Algebra</p> <p>* Network analysis: centrality, rating and ranking, similarity, community, structure</p> <p>* Visualization of networks</p> <p>Capacità relative alle discipline</p> <p><i>Conoscenza e capacità di comprensione:</i> lo studente dovrà aver acquisito le conoscenze e le capacità necessarie per analizzare una rete in un dominio qualsiasi</p> <p><i>Conoscenza e capacità di comprensione applicate:</i> lo studente dovrà essere in grado di usare l'ambiente R (in particolare il pacchetto igraph) e Processing per l'analisi e la visualizzazione di reti.</p> <p>Capacità trasversali/soft skills</p> <p><i>Autonomia di giudizio:</i> lo studente dovrà essere in grado di interpretare i risultati sperimentali e trarre conclusioni attinenti al dominio del discorso</p> <p><i>Abilità comunicative:</i> lo studente dovrà essere in grado di comunicare in modo efficace i risultati di una analisi sperimentale</p> <p><i>Capacità di apprendere:</i> lo studente dovrà dimostrare di aver appreso la tecnica di analisi delle reti e di saperla utilizzare in un contesto qualsiasi.</p>	<p>* The origins of networks and networks in the contemporary world</p> <p>* Introduction to Graph Theory and Linear Algebra</p> <p>* Network analysis: centrality, rating and ranking, similarity, community, structure</p> <p>* Visualization of networks</p> <p>Sector-specific skills</p> <p><i>Knowledge and understanding:</i> the student must have acquired the necessary knowledge to analyze a network in any domain</p> <p><i>Applying knowledge and understanding:</i> the student must have learned the R environment (in particular the igraph package) and the language Processing for analysis and visualization of networks.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>Making judgments:</i> the student must be able to interpret the experimental results and draw conclusions relevant to the domain of discourse</p> <p><i>Communication skills:</i> the student must be able to communicate effectively the results of an experimental analysis</p> <p><i>Learning skills:</i> the student must demonstrate that they have learned the technique of network analysis and be able to apply it in any domain.</p>	
Semantica e concorrenza	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</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,</p>	

	<p>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, non-determinismo, concorrenza; conoscere le principali problematiche dei sistemi concorrenti e distribuiti.</p> <p>1.2 <i>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>2.1 <i>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>2.2 <i>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>2.3 <i>Capacità di apprendimento</i>: il corso fornisce le nozioni base di metodi formali necessarie per poter apprendere tecniche di verifica più sofisticate.</p>	<p>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 programming languages; to know how the various aspects of computation are modeled: higher-order, non-determinism, concurrence; to know the main issues of concurrent and distributed systems.</p> <p>1.2 <i>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>2.1. <i>Making judgments</i>: students learn to be aware of the ambiguities present in informal descriptions, and identify the most appropriate formalization techniques.</p> <p>2.2 <i>Communication skills</i>: students learn to formalize and describe behavior from the various constructs of a programming language.</p>	
--	---	--	--

			2.3 <i>Learning skills</i> : the course provides the basic notions of formal methods needed to learn more sophisticated verification techniques.	
Cibersecurity	INF/01	<p>Lo/la studente/essa dovrà:</p> <ul style="list-style-type: none"> -conoscere i concetti fondamentali delle tecniche e degli algoritmi per la trasmissione sicura dei dati multimediali (immagini, video, tracce audio, etc.) attraverso la rete internet e, più in generale, attraverso canali di comunicazione non sicuri. -conoscere le principali tecniche di steganografia, watermarking e fingerpring -saper analizzare e comprendere gli elementi fondamentali delle tecniche di crittografia sia a chiave simmetrica (o privata) che a chiave asimmetrica (o pubblica). -saper utilizzare i principali linguaggi di programmazione per il WEB quali PHP, Java, MySQL e HTML5 -saper progettare e sviluppare un'applicazione multimediale sicura. <p>Capacità relative alle discipline:</p> <p><i>-Conoscenza e comprensione</i></p> <p>Acquisire specifiche conoscenze dei principali concetti e principi teorici della sicurezza dei sistemi multimediali, della crittografia, della staganografia di dati multimediali e delle tecniche di watermarking e fingerprint. Conoscere e saper utilizzare gli aspetti di sicurezza legati ai principali linguaggi di programmazione per il Web utilizzati nel settore dei sistemi multimediali e dei nuovi media digitali.</p>	<p>The student will have to:</p> <ul style="list-style-type: none"> -know the basic concepts of techniques and algorithms for the secure transmission of multimedia data (images, video, audio tracks, etc.) through the Internet and, more generally, through insecure communication channels. -know the main techniques of steganography, watermarking and fingerprint -know how to analyze and understand the key elements of encryption techniques either with symmetric and asymmetric keys. -know how to use the main programming languages for WEB applications such as PHP, Java, MySQL and HTML5 -know how to design and develop a secure multimedia application. <p>Sector-specific skills:</p> <p><i>-Knowledge and understanding</i></p> <p>Acquire specific knowledge of key concepts and theoretical principles of multimedia system security, cryptography, steganography, watermarking and fingerprint techniques.</p> <p>Know and know how to use the security aspects of the major programming languages for the Web used in the multimedia and digital media sectors.</p> <p><i>-Applying knowledge and understanding</i></p>	<p>Costituiscono prerequisiti del corso la conoscenza della matematica di base (in particolare, logaritmi, esponenziali, derivate, integrali, successioni aritmetiche e geometriche, funzioni, aritmetica modulare), dei sistemi operativi, delle reti di calcolatori (architetture di rete, protocolli, sistemi client/server, etc.) e dei sistemi multimediali.</p> <p>Basic element of mathematics (in particular, logarithms, exponential, derivative, integral, arithmetic and geometric sequences, modular functions and operators), operating systems, and computer networks (network architectures, protocols, client/server systems,</p>

		<p><i>-Capacità di applicare conoscenza e comprensione</i> Saper applicare a casi reali le conoscenze maturate sui diversi metodi di analisi e di modellazione di problemi relativi alla sicurezza dei dati generati e trasmessi dalle applicazioni multimediali. Saper applicare le conoscenze acquisite sulla trasmissione sicura di dati e informazioni alla progettazione e collaudo di sistemi multimediali (piattaforme cloud, social media, database distribuiti, sistemi interattivi, sistemi robotici, etc.), adattandoli a determinati contesti reali e a specifici requisiti di funzionamento.</p> <p>Capacità trasversali /soft skills <i>-Autonomia di giudizio</i> Saper valutare e definire autonomamente il livello di sicurezza dei sistemi multimediali. Saper identificare i problemi relativi alla sicurezza ed operare scelte sulle tecnologie dell'informazione più appropriate per risolverli. Saper valutare e scegliere gli algoritmi di crittografia (simmetrica o asimmetrica) più appropriati per la trasmissione di dati multimediali su differenti canali di comunicazione, identificandone effetti positivi e criticità. <i>-Abilità comunicative</i> Saper illustrare e presentare con rigore metodologico relazioni tecnico-scientifiche sulla realizzazione e sul funzionamento di sistemi informatici che prevedano la trasmissione sicura di dati multimediali attraverso la rete. <i>-Capacità di apprendimento</i></p>	<p>Know how to apply in real cases the acquired knowledge about the methods of analysis and modeling of security problems in multimedia applications. Know how to apply knowledge on secure data and information transmission to the design and testing of multimedia systems (cloud platforms, social media, distributed databases, interactive systems, robotic systems, etc.), adapting them to specific contexts and specific operating requirements.</p> <p>Cross-sectoral skills/soft skills <i>-Making judgments</i> Know how to independently assess and define the security level of multimedia systems. Know how to identify security issues and make choices about the most appropriate information technologies to solve them. Know how to evaluate and choose the most appropriate symmetric or asymmetric encryption algorithms for the transmission of multimedia data over different communication channels, identifying its positive and critical effects. <i>-Communication Skills</i> To illustrate and organize rigorous technical-scientific relations on the design and functioning of computer systems that involve the transmission of multimedia data through the network. <i>-Learning skills</i> Know how to find and use appropriate bibliographic and technology tools to learn</p>	<p>etc.) are prerequisites of the course.</p>
--	--	--	--	---

		Saper reperire e utilizzare appropriati strumenti bibliografici e tecnologici utili per l'apprendimento autonomo di problemi inerenti le tecniche e/o i protocolli per la sicurezza più adatti alla trasmissione di dati multimediali su canali insicuri.	autonomously security techniques and protocols most suited to the transmission of multimedia data to insecure channels.	
Sistemi distribuiti	INF/01	<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>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 <i>Conoscenza e capacità di comprensione:</i> lo studente acquisisce le conoscenze e capacità necessarie per analizzare e progettare un sistema distribuito, e comprendere eventuali problematiche ad esso connesse. Le conoscenze</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> <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</p>	

	<p>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><i>1.2 Conoscenza e capacità di comprensione applicate:</i> 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>Capacità trasversali/soft skills</p> <p><i>2.1 Autonomia di giudizio:</i> 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><i>2.2 Abilità comunicative:</i> 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><i>2.3 Capacità di apprendimento:</i> 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>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><i>1.2 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 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><i>2.1. 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><i>2.2 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>	
--	--	---	--

			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.	
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>Lo studente dovrà:</p> <p>Capacità relative alle discipline</p> <p>3. <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>4. <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,</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 consideration also technology and supply relationships.</p> <p>The student will have to:</p> <p>Sector-specific skills</p> <p>3. <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>4. <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</p>	

		<p>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>4. <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>5. <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>6. <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>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>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 computer data and business processes definition.</p> <p>5. <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>6. <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>	
Tecnologie Web avanzate	ING-INF/05	<p>Il corso si propone di introdurre gli studenti alle Tecnologie Web destinate all'interoperabilità, alla condivisione di risorse ed alla cooperazione, sia dal punto di vista concettuale che pratico tramite esercitazioni e casi di studio. Il corso consente allo studente di completare la formazione in tema di tecnologie web già iniziata con i corsi di Tecnologie Web e Complementi di Tecnologie Web.</p>	<p>The present course is aimed at introducing students to web technologies related to interoperability, resource sharing, and cooperation, from a conceptual as well as practical point of view, the latter by means of case studies and exercises. The course allows the student to complete the education on web technologies started</p>	<p>Programmazione, programmazione orientata agli oggetti, tecnologie web lato client e lato server. Programming and object-oriented programming, client- and server-side web technologies.</p>

	<p>Lo/la studente/essa dovrà:</p> <p>Capacità relative alle discipline</p> <p><i>1.1. Conoscenza e capacità di comprensione</i> Conoscere i principi della Service Orientation Conoscere le caratteristiche specifiche ed i protocolli relativi ai servizi Web di tipo SOAP e di tipo REST, con attenzione al formato JSON Conoscere i principi del Cloud ed i tipi di servizi forniti dai cloud provider Comprendere le problematiche di base nella realizzazione di applicazioni orientate alle piattaforme di social media.</p> <p><i>1.2 Capacità di applicare conoscenza e comprensione</i> Saper implementare un'applicazione che utilizza servizi web esterni Saper implementare un provider di servizi Web Saper utilizzare alcuni servizi cloud nelle proprie applicazioni</p> <p>Capacità trasversali / soft skills</p> <p><i>2.1 Autonomia di giudizio</i> Saper selezionare le librerie ed i servizi web necessari per lo sviluppo di un determinato progetto Saper identificare le funzioni da fornire tramite web services Saper identificare quali, tra le risorse fornite dai vari cloud provider, siano le più adeguate alla soluzione di un determinato problema implementativo, sia dal punto di vista delle funzionalità che dei costi.</p> <p><i>2.2 Abilità comunicative.</i> Essere in grado di motivare, a voce e per iscritto, le scelte implementative effettuate nello</p>	<p>with the previous courses on Web Technologies.</p>	
--	--	---	--

		<p>sviluppo di un progetto basato su tecnologie lato server, web services o cloud.</p> <p><i>2.3 Capacità di apprendimento</i></p> <p>Saper reperire ed utilizzare risorse informatiche e scientifiche per l'approfondimento autonomo delle tematiche studiate a lezione</p> <p>Saper affrontare lo studio di nuove tecnologie web.</p> <p>Saper individuare nuovi servizi web, inclusi quelli di tipo cloud, che possano essere utili per lo sviluppo di applicazioni di interesse</p>		
Verifica automatica dei sistemi: teoria e applicazioni	INF/01	<p>Il corso vuole presentare in modo sistematico metodi, formalismi e algoritmi per la specifica formale e la verifica automatica di sistemi reattivi complessi. 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.</p> <p>Ampio spazio verrà dedicato alla teoria degli automi operanti su oggetti infiniti (parole e alberi) e alle logiche temporali comunemente utilizzate (LTL, CTL, CTL*). Particolare attenzione verrà riservata ai risultati relativi all'equivalenza espressiva tra classi di automi e sistemi logici. Verrà, inoltre, illustrato il possibile utilizzo della teoria logica dei giochi nella verifica formale. Dal punto di vista algoritmico, con riferimento ai modelli computazionali e ai formalismi di specifica presi in considerazione, verranno</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. 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.</p> <p>A prominent role is assigned to the theory of automata operating on infinite objects (words and trees) and to the most commonly used temporal logics (LTL, CTL, 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</p>	

	<p>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>Capacità relative alle discipline</p> <p><i>1.1 Conoscenza e capacità di comprensione:</i> 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><i>1.2 Capacità di applicare conoscenza e comprensione:</i> lo studente</p>	<p>in formal verification is illustrated as well. 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 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><i>1.1 Knowledge and understanding:</i> 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)</p>	
--	--	--	--

	<p>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><i>2.1 Autonomia di giudizio:</i> 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><i>2.2 Abilità comunicative:</i> 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><i>2.3 Capacità di apprendimento:</i> la natura degli argomenti trattati nel corso, le modalità di organizzazione e presentazione dei diversi</p>	<p>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><i>1.2 Applying knowledge and understanding:</i> 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 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><i>2.1 Making judgements:</i> 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><i>2.2 Communication skills:</i> the writing of the technical report will be followed by an</p>	
--	---	---	--

		<p>contenuti e le modalità di verifica della loro effettiva comprensione sono finalizzati alla crescita delle capacità di apprendimento degli studenti.</p>	<p>oral presentation of its main contents. All students attending the course will be invited at such a presentation. Such an activity aims at 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><i>2.3 Learning skills:</i> 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>	
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 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</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 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</p>	

	<p>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>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>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 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</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>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>Architetture di sistemi applicativi: esempi</p> <p>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</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><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>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><i>1.2 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><i>2.1 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><i>2.2 Abilità comunicative.</i> Essere in grado di motivare per scritto, in un documento tecnico, le scelte di modellazione concettuale e 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</p>	<p><i>1.1 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><i>1.2 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><i>2.1 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 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</p>	
--	--	--	--	--

		soluzioni per il WS: standard tecnologici, ambienti di sviluppo.	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.	
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> <p>5. <i>Conoscenza e capacità di comprensione:</i> acquisire specifiche conoscenze dei principali concetti e principi dell'IA. Conoscere e sapere utilizzare le tecniche</p>	<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>	

		<p>base per la rappresentazione della conoscenza e dell'apprendimento automatico, anche mediante specifiche esercitazioni laboratoriali</p> <p>6. <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 conoscenze di cui sopra in specifici contesti applicativi.</p> <p>Capacità trasversali / soft skills Lo studente dovrà:</p> <p>7. <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>8. <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>9. <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>5. <i>Knowledge and understanding</i>: acquiring specific knowledge of the main concepts and basic principles of AI. Knowing and exploiting basic techniques for knowledge 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>6. <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>7. <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>8. <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>	
--	--	---	--	--

			9. <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.	
Metodi formali e logica per l'informatica - Modulo 1	MAT/01	<p>Il corso vuole fornire un panorama del ruolo della Logica nelle sue aree d'intersezione con l'Informatica, in particolare nella Teoria dei Giochi, nella Teoria degli Automi e nell'Intelligenza Artificiale.</p> <p>Il corso è diviso in tre parti. Nella prima parte si analizzano i giochi di Ehrenfeucht. Obiettivo di questa parte è familiarizzare lo studente con la teoria dei giochi e con il concetto di strategia vincente e mostrare le applicazioni della teoria dei giochi allo studio dell'espressività della logica al prim'ordine. Nella seconda parte si considerano automi a stati finiti su parole infinite utilizzandoli per dimostrare risultati classici sulla decidibilità della logica monadica al second'ordine. Obiettivo di questa parte è familiarizzare lo studente con gli automi che leggono parole infinite e con il loro potere espressivo. Nell'ultima parte si introducono e studiano logiche per il ragionamento non monotono. Obiettivo di questa parte è di mostrare come in alcuni ambiti la logica classica risulti inadeguata e possa essere sostituita da logiche non monotone.</p> <p>Capacità relative alle discipline</p> <p><i>1.1 Conoscenza e capacità di comprensione</i> Conoscere le analogie e le differenze in potere espressivo delle logiche introdotte nel corso e essere in grado di interpretare tali logiche utilizzando giochi ed automi. Essere in grado di</p>	<p>This course aims to provide a landscape of the role of Logic in Computer Science. The course is divided into three parts. In the first part Ehrenfeucht games are analyzed. The objective here is to get the student acquainted with game theory and to the concept of winning strategy, and to show the applications of game theory to the expressive power of first order logic. In the second part, finite state automata reading infinite words are introduced and used to prove some classical results on monadic second order logic. The objective here is to get the student acquainted with automata reading infinite words and with their expressive power. In the last part, non-monotone logics are introduced and studied. The objective here is to show that there are settings where classical logic is not adequate and can be replaced by non-monotone logic.</p> <p>Sector-specific skills</p> <p><i>1.1 Knowledge and understanding</i> To know analogies and differences in expressive power for the logics introduced in the course, and to be able to analyze such logics using games and automata. To be able to formalize non-monotone reasoning using an appropriate logic.</p> <p><i>1.2 Applying knowledge and understanding</i></p>	

		<p>formalizzare un ragionamento non monotono utilizzando una logica appropriata.</p> <p><i>1.2 Conoscenza e capacità di comprensione applicate</i></p> <p>Saper riconoscere la logica più adatta alla formalizzazione di una specifica proprietà, ed essere in grado di formalizzare correttamente tale proprietà.</p> <p>Saper dimostrare che una data proprietà non è esprimibile nella logica al prim'ordine.</p> <p>Capacità trasversali/soft skills</p> <p><i>2.1 Autonomia di giudizio</i></p> <p>Saper riconoscere una formalizzazione corretta di un dato problema.</p> <p>Saper valutare quale strumento utilizzare per formalizzare uno specifico problema.</p> <p><i>2.2 Abilità comunicative.</i></p> <p>Saper comunicare le proprie intuizioni e dimostrazioni ai compagni di corso.</p> <p>Riuscire a formalizzare un problema nel modo più semplice possibile e riuscire a spiegare tale formalizzazione in modo comprensibile.</p> <p>Scrivere in modo formalmente corretto la soluzione degli esercizi.</p> <p><i>2.3 Capacità di apprendimento</i></p> <p>Essere in grado di approfondire in maniera autonoma gli argomenti trattati nel corso.</p>	<p>To be able to recognize the logic which is more suited for formalizing a specific property and to perform such a formalization correctly.</p> <p>Cross-sectoral skills/soft skills</p> <p><i>2.1 Making judgements</i></p> <p>To be able to recognize a correct formalization.</p> <p>To be able to evaluate which tool is more suited to solve a specific problem.</p> <p><i>2.2 Communication skills</i></p> <p>To know how to communicate the intuition behind a formalization of a problem to classmates. 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><i>2.3 Learning skills</i></p> <p>To be able to further develop and deepen, by independent study, the knowledge of the role of logic</p>	
Metodi formali e logica per l'informatica - Modulo 2	INF/01	<p>L'obiettivo del modulo 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</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 program invariants, will be presented for the verification of sequential, parallel and distributed programs. Correctness and termination of a number of classical</p>	<p>Programmazione e laboratorio Logica Matematica Sistemi Operativi. Programming Logic Operating Systems</p>

		<p>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>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.</p>	
Ricerca operativa	MAT/09	<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).</p> <p>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>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>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>
Statistica applicata e analisi dei dati	SECS-S/01	<p>Capacità relative alla disciplina</p> <p><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.</p> <p><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</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.</p> <p>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</p>	

		<p>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</p> <p><i>Autonomia di giudizio:</i> autonomia di giudizio nella scelta dei modelli e dei metodi statistici più 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>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</p> <p><i>1.1 Knowledge and understanding</i> 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> 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> Making judgements on the appropriate statistical models and methods to be used</p>	
--	--	--	--	--

			<p>for analyzing a specific dataset and on the interpretation of the experimental results.</p> <p><i>2.2 Communication skills</i> 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>	
--	--	--	---	--