



**UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II**  
**SCUOLA POLITECNICA E DELLE SCIENZE DI BASE**

**DIPARTIMENTO DI INGEGNERIA ELETTRICA E  
DELLE TECNOLOGIE DELL'INFORMAZIONE**

**GUIDA DELLO STUDENTE**

**CORSO DI LAUREA MAGISTRALE IN  
INGEGNERIA INFORMATICA**

*Classe delle Lauree Magistrali in Ingegneria Informatica- LM-32*

**ANNO ACCADEMICO 2016/2017**

**Napoli, luglio 2016**

## **Finalità del Corso di Studi e sbocchi occupazionali**

Obiettivo del Corso di Laurea magistrale in Ingegneria Informatica è quello di formare un professionista in grado di inserirsi in realtà produttive molto differenziate e in rapida evoluzione con ruoli di promozione e gestione dell'innovazione tecnologica, di progetto e di gestione di sistemi complessi, di coordinamento di gruppi di lavoro e di responsabilità in ambito tecnico e produttivo ai massimi livelli. Oltre agli ambiti specifici dell'Ingegneria Informatica, le sue competenze copriranno anche altri ambiti dell'Ingegneria con particolare riferimento all'intero settore delle Tecnologie dell'Informazione e della Comunicazione.

La formazione professionale del laureato magistrale in Ingegneria Informatica richiede pertanto l'acquisizione di capacità progettuali avanzate e con contenuti innovativi sia nell'area delle architetture dei sistemi di elaborazione, sia in quella dei sistemi software sia in quella delle applicazioni e dei sistemi telematici. Oltre alle conoscenze di tipo specificamente professionale e tecnologico, il laureato magistrale in Ingegneria Informatica deve possedere una ampia e solida formazione sugli aspetti teorico-scientifici della matematica e delle altre scienze di base, nonché sugli aspetti teorico-scientifici dell'ingegneria, sia in generale sia in modo approfondito relativamente a quelli dell'ingegneria informatica. Egli deve essere capace di utilizzare tale conoscenza per interpretare e descrivere problemi dell'ingegneria complessi o che richiedono un approccio interdisciplinare.

La formazione del laureato magistrale in Ingegneria Informatica deve infine comprendere la conoscenza in forma scritta e orale, di almeno una lingua dell'Unione Europea oltre l'italiano, con riferimento anche ai lessici disciplinari. La figura professionale trova significative prospettive di occupazione in enti pubblici e privati, in società di ingegneria e in imprese manifatturiere, operanti negli ambiti della produzione hardware e software, nell'area dei sistemi informativi e delle reti di calcolatori, nelle imprese di servizi, nei servizi informatici della pubblica amministrazione, e ovunque sia presente il problema dell'elaborazione e della gestione dell'informazione.

**Manifesto del Corso di Laurea magistrale in Ingegneria Informatica**  
 (Classe delle Lauree Magistrali in Ingegneria Informatica, Classe LM-32)  
 A.A. 2016/2017

Insegnamento o attività formativa	CFU	SSD	Ambito Disciplinare	Tipologia (*)	Propedeuticità
<b>I Anno – I semestre</b>					
Architettura dei sistemi di elaborazione	9	ING-INF/05	Ingegneria Informatica	2	
Algoritmi e strutture dati	6	ING-INF/05	Ingegneria Informatica	2	
Trasmissione numerica	6	ING-INF/03	Attività formative affini o integrative	4	
<b>I Anno – II semestre</b>					
Calcolatori elettronici II	6	ING-INF/05	Ingegneria Informatica	2	
Ricerca operativa	9	MAT/09	Attività formative affini o integrative	4	Algoritmi e strutture dati
Calcolo numerico	9	MAT/08	Attività formative affini o integrative	4	
Progettazione e sviluppo di sistemi software	9	ING-INF/05	Ingegneria Informatica	2	
<b>II Anno – I semestre</b>					
Impianti di elaborazione	9	ING-INF/05	Ingegneria Informatica	2	
Attività formative curriculari a scelta dello studente (vedi nota a)	12	ING-INF/05	Ingegneria Informatica	2	
<b>II Anno – II semestre</b>					
Attività formative curriculari a scelta dello studente (vedi nota a)	6	ING-INF/05	Ingegneria Informatica	2	
Attività formative a scelta dello studente (vedi nota b)	15		A scelta dello studente	3	
Altre attività formative	12		Ulteriori attività formative	6	
Prova finale	12		Ulteriori attività formative	5	

- a) Lo studente dovrà scegliere una delle aree presenti in **Tabella A**.
- b) I CFU di tipologia 3 possono essere usufruiti per intero nel secondo semestre, ovvero in parte nel primo e in parte nel secondo semestre. Lo studente potrà attingere, tra l'altro, ad attività formative indicate in **Tabella B**.

**Tabella A. Attività formative curriculari a scelta dello studente**

Area	Insegnamento o Attività formativa	CFU	SSD	Sem.	Ambito Disciplinare	Tipologia (*)	Propedeuticità
<b>IT Management</b>	Sistemi Informativi	6	ING-INF/05	I	Ingegneria Informatica	2	
	Data Mining	6	ING-INF/05	I	Ingegneria Informatica	2	
	Big Data Analytics and Business Intelligence	6	ING-INF/05	II	Ingegneria Informatica	2	
<b>Sistemi Embedded ed Industriali</b>	Sistemi Distribuiti	6	ING-INF/05	I	Ingegneria Informatica	2	
	<i>Secure Systems Design</i>	6	ING-INF/05	I	Ingegneria Informatica	2	
	<i>Sistemi embedded</i>	6	ING-INF/05	II	Ingegneria Informatica	2	Architettura dei sistemi di elaborazione Calcolatori elettronici II
<b>Reti ed Internet</b>	Computer Networks II	6	ING-INF/05	I	Ingegneria Informatica	2	
	Applicazioni telematiche	6	ING-INF/05	I	Ingegneria Informatica	2	
	Protocolli per reti mobili	6	ING-INF/05	II	Ingegneria Informatica	2	

**Tabella B. Attività formative disponibili per la scelta autonoma dello studente**

Insegnamento o Attività formativa	CFU	SSD	Sem.	Ambito Disciplinare	Tipologia (*)	Propedeuticità
Ingegneria del software II (1)	6	ING-INF/05	I	A scelta dello studente	3	
<i>Sistemi di Information Retrieval</i> (1)	6	ING-INF/05	II	A scelta dello studente	3	
<i>Automazione dei Processi di Business</i> (1)	3	ING-INF/05	II	A scelta dello studente	3	
Semantic Web (1)	6	INF/01	I	A scelta dello studente	3	
Economia ed Organizzazione Aziendale (1)	9	ING-IND/35	II	A scelta dello studente	3	
Metodi per le decisioni – Ricerca Operativa II (1)	9	MAT/09	I	A scelta dello studente	3	
<i>Advanced Computer Architecture and GPU Programming</i> (2)	6	ING-INF/05	I	A scelta dello studente	3	
Sistemi real-time (2)	6	ING-INF/05	II	A scelta dello studente	3	
Metodi formali (2)	3	ING-INF/05	II	A scelta dello studente	3	
Specifica dei Sistemi (2)	6	INF/01	II	A scelta dello studente	3	
Circuiti per DSP (2)	9	ING-INF/01	I	A scelta dello studente	3	
Architettura dei Sistemi Integrati (2)	9	ING-INF/01	II	A scelta dello studente	3	
<i>Network Security</i> (3)	6	ING-INF/05	I	A scelta dello studente	3	
<i>Analisi e Prestazioni di Internet</i> (3)	6	ING-INF/05	II	A scelta dello studente	3	
<i>Cloud e Datacenter Networking</i> (3)	3	ING-INF/05	II	A scelta dello studente	3	
Calcolo parallelo (3)	6	MAT/08	I	A scelta dello studente	3	
Reti wireless (3)	9	ING-INF/03	I	A scelta dello studente	3	
Ottimizzazione (3)	9	MAT/09	I	A scelta dello studente	3	

(1) Insegnamenti suggeriti per coloro che scelgono l'area curriculare *IT Management*

(2) Insegnamenti suggeriti per coloro che scelgono l'area curriculare *Sistemi Embedded ed Industriali*

(3) Insegnamenti suggeriti per coloro che scelgono l'area curriculare *Reti ed Internet*

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

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

## Attività formative del Corso di Studi

<b>Insegnamento: Algoritmi e strutture dati</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 36	<b>Ore di esercitazione:</b> 12
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b> Fornire gli strumenti necessari per la sintesi e l'analisi di algoritmi e strutture dati anche complessi. Le capacità di sintesi verranno sviluppate attraverso lo studio di una ampia varietà di strutture dati e di algoritmi che risolvono problemi di carattere fondamentale nello sviluppo delle applicazioni informatiche. Per quanto riguarda le capacità di analisi, verranno introdotte le tecniche di base per la dimostrazione di correttezza di un algoritmo e la valutazione della complessità temporale.	
<b>Contenuti:</b> <u>Concetti introduttivi:</u> algoritmi e strutture dati, ricorsione, divide-et-impera. <u>Analisi di correttezza:</u> invariante di ciclo, correttezza di algoritmi ricorsivi. <u>Analisi di complessità:</u> le notazioni asintotiche $O$ , $\Omega$ , $\Theta$ ; analisi di algoritmi ricorsivi. <u>Algoritmi di ordinamento e statistiche d'ordine:</u> mergesort, heapsort, quicksort, ordinamento in tempo lineare, mediane e statistiche d'ordine. <u>Strutture dati:</u> pile e code; code di priorità; liste; tabelle hash; alberi binari di ricerca, RB-alberi, alberi splay. Grafi. <u>Tecniche avanzate di progettazione e di analisi:</u> programmazione dinamica, algoritmi golosi, analisi ammortizzata. <u>Esempi di algoritmi per la soluzione di problemi specifici:</u> DES ed RSA. <u>Traduttori e interpreti:</u> analisi lessicale, analisi sintattica, analisi semantica, interpreti, strutture dati usate nei traduttori.	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b> nessuna	
<b>Metodo didattico:</b>	
<b>Materiale didattico:</b> T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein: Introduzione agli algoritmi e strutture dati 3/ed. Mc-GrawHill Italia, 2010; Dispense didattiche.	
<b>Modalità di esame:</b> Sviluppo di un elaborato, prova scritta e prova orale.	

<b>Insegnamento: Applicazioni telematiche</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso si propone di fornire le nozioni teoriche e metodologiche di base per la progettazione e lo sviluppo di applicazioni telematiche, con particolare riferimento ai sistemi basati sul web ed alle applicazioni multimediali distribuite. Le applicazioni telematiche verranno studiate sia dal punto di vista dell'architettura software che dal punto di vista dei protocolli che definiscono le modalità di comunicazione. Il corso si articola in tre parti: 1) Progetto e sviluppo di applicazioni basate sul web; 2) Progetto e sviluppo di applicazioni multimediali distribuite; 3) Paradigmi di comunicazione alternativi per applicazioni telematiche. La presentazione degli aspetti teorici è integrata da un'attività di esercitazione in laboratorio.	
<b>Contenuti:</b> <u>Parte I:</u> Applicazioni basate sul web. Interazione client-server nel Web. Il Protocollo HTTP. Web caching e problematiche connesse. Web Server. Servizio di pagine statiche. Applicazioni Web dinamiche: programmazione server-side. Linguaggi di scripting per il web. Applicazioni Web in Java: servlet e Java Server Pages (JSP). XML come formato di interscambio dati. Validazione e parsing di documenti XML. Dalle applicazioni Web ai Web Services. Service Oriented Architectures (SOA). Lo stack protocollare dei Web Services. <u>Parte II:</u> Applicazioni Multimediali Distribuite. Protocolli a supporto dello streaming di flussi audio/video. Il protocollo RTP. Il protocollo RTSP per il controllo di sessioni. Protocolli di segnalazione per telefonia su IP: SIP. Applicazioni di video-on-demand e conferencing in Internet. Realizzazione di applicazioni di telefonia su IP: le SIP servlet. Applicazioni convergenti HTTP/SIP. Applicazioni di Instant Messaging: il protocollo XMPP. <u>Parte III:</u> Paradigmi di comunicazione alternativi per applicazioni telematiche. Dal modello client-server al modello peer-to-peer. Architettura delle applicazioni peer-to-peer. Applicazioni di file sharing. Il peer-to-peer all'interno dei browser web: attività di standardizzazione in ambito W3C (WebRTC – Web Real-Time Communications) e IETF (RtcWeb – Real-Time Communication in WEB browsers). Dalla comunicazione sincrona alla comunicazione asincrona: comunicazione mediante code di messaggi; comunicazione secondo il modello publish-subscribe; comunicazione bidirezionale tramite il protocollo HTTP.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti:</b> Nessuno	
<b>Metodo didattico:</b> lezioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> appunti del corso, articoli scientifici, documenti standard per Internet (RFC). Testo consigliato per gli argomenti legati alla comunicazione peer-to-peer nei browser web: S. Loreto and S. P. Romano, "Real-Time Communication with WebRTC", O'Reilly Media, April 2014 (Ebook), May 2014 (Print) Print ISBN:978-1-4493-7187-6   ISBN 10:1-4493-7187-6 Ebook ISBN:978-1-4493-7182-1   ISBN 10:1-4493-7182-5	
<b>Modalità di esame:</b> progetto, realizzazione e discussione di un elaborato; colloquio orale sugli argomenti del corso.	

<b>Insegnamento: Architettura dei Sistemi di Elaborazione</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 52	<b>Ore di esercitazione:</b> 20
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b>	
<p>Il corso si pone l'obiettivo di fornire un'impostazione metodologica sul tema dell'architettura dei calcolatori. Prevede lo studio di metodi e tecniche per la progettazione dei sistemi di elaborazione dedicati e general purpose, con particolare riferimento all'architettura, all'organizzazione dei sistemi calcolatori a microprocessore e al progetto di unità di I/O.</p> <p>Il corso fa ampio riferimento agli argomenti già trattati nei corsi di Fondamenti di Informatica, Programmazione I e Calcolatori Elettronici I</p>	
<b>Contenuti:</b>	
<p>I sistemi digitali general purpose, special purpose e embedded. Progettazione dei sistemi digitali: aspetti tecnologici, metodologici e ambienti a supporto. Il ciclo di sviluppo di un sistema digitale. Richiami di reti logiche. Realizzazioni di funzioni booleane mediante circuiti digitali (ritardi, deformazioni dei segnali, dissipazioni potenze, ecc). Minimizzazione funzioni booleane: metodo McCluskey per funzioni mono e più uscite; minimizzazione a più livelli di reti combinatorie; metodi di rappresentazione; modello algebrico; trasformazioni algebriche e booleane; valutazione dei ritardi. Approfondimenti delle macchine sequenziali: macchine impulsive, a livelli, sincrone, asincrone. Sistemi di reti sequenziali: determinazione del tempo di ciclo; reti sequenziali a catena aperta e chiusa; architettura a pipeline. Dispositivi per la sintesi delle reti logiche: porte logiche, PAL, PLA, FPGA, Gate array. Sintesi automatica e mapping tecnologico. Sistemi digitali; funzioni e struttura dei sistemi; micro operazioni e loro descrizione; componenti logici; operazioni del sistema; sistemi digitali complessi (modello PO/PC); sottosistema di controllo e di calcolo (datapath); Sistemi a controllo cablato e micro programmato; tipologia dei micro linguaggi; corrispondenza fra microlinguaggi ed automi. Le macchine aritmetiche: addizionatori, sottrattori, moltiplicatori e divisori binari e decimali. Algoritmi per la moltiplicazione e algoritmo di Booth. Algoritmi per la divisione: restoring e non restoring. Aritmetica in virgola mobile e algoritmi fondamentali per il trattamento. Progettazione dei circuiti aritmetici e sintesi su PAL e FPGA. I componenti di un datapath: registri, circuiti di selezione, multiplexer/demultiplexer, decoder, shifter. Le memorie: ROM, PROM, E2PROM e flash; RAM statiche e dinamiche. Architetture e progettazione dei datapath; Processori general purpose CISC/RISC e loro repertorio dei codici operativi. I sistemi di I/O. Comunicazione Seriale e Parallela. Architettura di un sistema embedded ed esempi di impiego. I sensori e gli attuatori. I linguaggi HDL per la descrizione dell'hardware e gli ambienti di simulazione. Il linguaggio VHDL. Nelle ore di laboratorio lo studente dovrà sintetizzare su una scheda con FPGA i circuiti proposti e/o discussi nelle ore di esercitazione</p>	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni frontali, esercitazioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> slides del corso, libri di testo	
<b>Modalità di esame:</b> realizzazione e discussione di un progetto e prova orale.	



<b>Insegnamento: Big Data Analytics e Business Intelligence</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Scopo del corso è fornire i fondamenti dei sistemi di Business Intelligence, dei Big Data e lo stato dell'arte dell'analytics. Il corso in particolare fornisce una introduzione ai Big Data e Data Analytics Lifecycle, con riferimento alla progettazione di sistemi di dati di grandi dimensioni e complessi, e ai processi di modellizzazione, acquisizione, curation, condivisione, analisi e visualizzazione delle informazioni presenti nei Big Data. Durante il corso, inoltre, saranno presentate tecnologie e tools per la gestione dei big data, fornendo allo studente le conoscenze necessarie e le applicazioni pratiche per l'applicazione dei big data alla cosiddetta X-Informatics.	
<b>Contenuti:</b> Business Intelligence (BI): introduzione e principi fondamentali. Il ciclo di vita della BI. Architettura di un sistema di BI; progettazione di un Sistema di BI. Sistemi di business intelligence open source e commerciali. Introduzione ai Big Data: definizione di un sistema Big Data. Modelli dei dati per Big Data. Linked Data. Map Reduce e NoSQL. Key-value systems - Column-family storage systems, Graph storage systems, Document Database systems. Introduzione alla Big Data Analytics (BDA): stato dell'arte della BDA. BDA Lifecycle: knowledge discovery in database, data preparation, model planning, model building, data visualization. Esempi di Tools commerciali ed opensource: Oracle, SAS, IBM Business Analytics, Pentaho, R e Hadoop.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Sistemi Informativi. Data Mining	
<b>Metodo didattico:</b> Lezioni frontali ed esercitazioni in laboratorio.	
<b>Materiale didattico:</b> Appunti del docente. Lucidi. Articoli scientifici. Libri di testo consigliati: "Big Data: Architettura, tecnologie e metodi per l'utilizzo di grandi basi di dati", A. Rezzani, APOGEO, 2013. "Business intelligence. Modelli matematici e sistemi per le decisioni", C. Vercelli, MacGraw-Hill Companies, 2006	
<b>Modalità di esame:</b> Elaborato, prova orale.	

<b>Insegnamento: Calcolatori elettronici II</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 30	<b>Ore di esercitazione:</b> 18
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b>	
<p>Il corso affronta lo studio delle principali tecnologie e metodologie di progettazione dei sistemi di elaborazione dedicati (sistemi embedded ) e general purpose utilizzati in ambito industriale (avionica, meccanica, trasporti, chimica, ecc), con particolare riferimento all'architettura, all'organizzazione dei sistemi calcolatori a microprocessore e al progetto di unità di I/O. Il corso fa ampio riferimento agli argomenti già trattati nei corsi di Fondamenti di Informatica, Calcolatori Elettronici I, Sistemi Operativi e Architettura dei Sistemi di Elaborazione.</p>	
<b>Contenuti:</b>	
<p>Analisi degli aspetti metodologici, tecnologici e implementativi di processori appartenenti alle più diffuse famiglie di microprocessori RISC e CISC con particolare riferimento al Motorola 68000 e Sparc MIPS. Analisi e sviluppo di componenti da integrare in un sistema di elaborazione; tecniche di programmazione assembler per 68000 e MIPS (esempi di programmi assembler: gestione dei sottoprogrammi; gestione stack e code). Architettura dell'unità di calcolo: il datapath e la tempificazione delle micro-operazioni; l'interfaccia verso la memoria. Tecniche per l'aumento delle prestazioni di una CPU: Pipeline; esecuzione fuori ordine; parallelismo funzionale, processori superscalari. Architettura dell'unità di controllo: il modello PO/PC personalizzato alla realizzazione di un microprocessore. Implementazione architettura. Interruzioni hw e sw. Principali meccanismi di gestione delle interruzioni (abilitazione, disabilitazione, identificazione, salvataggio e ripristino dello stato, servizio, gestione prioritaria). Interruzioni precise nei processori superscalari. Il sistema memoria: gerarchia di memorie; tecnologie delle memorie; architetture delle memorie centrali e cache, il progetto del sistema memoria. Le operazioni di I/O. Il modello stato-controllo-dato di un sistema di I/O (device e periferica); modello astratto di un device. Metodi per la selezione di un device/periferica. Il sistema BUS. Protocolli di handshaking per il trasferimento dei dati. Driver per il controllo dei dispositivi di I/O; principali periferiche per microcomputer. Esempi di dispositivi per: comunicazioni seriali sincrone e asincrone, comunicazione parallela, l'accesso diretto alle memorie, la gestione prioritaria delle interruzioni. Sistemi multiprocessore gestione della memoria; Architettura dei sistemi multiprocessore (topologia di interconnessione reti dirette e indirette). Durante le ore di laboratorio lo studente imparerà a progettare e sviluppare driver per il controllo dei dispositivi di I/O, a dimensionare sistemi di memoria, a programmare in linguaggio assembler.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni frontali, esercitazioni	
<b>Materiale didattico:</b> slide dalle lezioni, libri di testo	
<b>Modalità di esame:</b> La verifica prevede una prova scritta e una prova orale.	

<b>Insegnamento: Calcolo Numerico</b>	
<b>CFU:</b> 9	<b>SSD:</b> MAT/08
<b>Ore di lezione:</b> 50	<b>Ore di esercitazione:</b> 22
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b> Scopo del corso è introdurre lo studente al calcolo scientifico, ossia allo sviluppo di metodologie, tecniche e competenze operative legate allo sviluppo di algoritmi e software che permettono di risolvere un problema scientifico tramite calcolatore. Il corso prevede lo studio del sistema MATLAB, utilizzato per lo sviluppo di software nei progetti applicativi degli studenti.	
<b>Contenuti:</b> <i>Introduzione al Calcolo Scientifico.</i> Modello matematico, modello numerico, algoritmo, software, fonti di errore. Sistema aritmetico Standard IEEE. Errore di round off. Criterio di arresto di un processo iterativo. Condizionamento e stabilità. Progettazione, valutazione e documentazione del software matematico. <i>Il sistema MATLAB.</i> Gestione, manipolazione e funzioni elementari di array. Variabili strutturate. Programmare in Matlab: costrutti di controllo, script files, function files, vettorializzazione del codice. Grafica in due e tre dimensioni, cenni di grafica avanzata e animazione. <i>Algebra Lineare Numerica.</i> Condizionamento di un sistema lineare. Sistemi triangolari. Algoritmo di Gauss, fattorizzazione LU e pivoting. Sistemi tridiagonali. Gestione delle matrici sparse. Sistemi lineari di grandi dimensioni: algoritmi di Jacobi, Gauss Seidel. Applicazioni: grafi, catene di Markov, ordinamento del WEB: Algoritmo PageRank di Google. <i>Fitting di dati.</i> Interpolazione con polinomi, polinomi a tratti, funzioni spline. Spline cubica not-a-knot, naturale e completa. Interpolazione di curve: spline parametrica. Polinomio di minimi quadrati. <i>Integrazione Numerica.</i> Formule di quadratura elementari e composite. Algoritmi di quadratura automatica, stima dell'errore, criterio di arresto. Algoritmo adattativo. <i>Risoluzione numerica di equazioni differenziali ordinarie.</i> Metodi di Eulero esplicito ed implicito. Stabilità e convergenza. Metodi Runge Kutta. Equazioni Stiff. Ode Suite del Matlab. Applicazioni: sviluppo di una popolazione, equazioni di Lotka-Volterra, problema del corpo che cade, equazione di Van der Pool, equazioni di Lorentz. <i>Trasformata discreta di Fourier.</i> Definizione e proprietà. Campionamento, frequenza di Nyquist, periodogramma. Aliasing. Esempi ed applicazioni in Matlab : filtraggio di un segnale, analisi di fenomeni periodici, analisi di segnali sonori. Algoritmo FFT radix-2.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni, laboratorio	
<b>Materiale didattico:</b> libri di testo: A.D'Alessio- <i>Lezioni di Calcolo Numerico e Matlab</i> - Liguori editore. II ed-2006. Dispense in rete	
<b>Modalità di esame:</b> prova scritta, colloquio con discussione sugli elaborati svolti.	

<b>Insegnamento: Computer Networks II</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:II</b>	
<b>Obiettivi formativi:</b> This course aims to provide advanced methodological and technological competences on the design and management of computer networks and complex telematics services. The educational objectives are to give: advanced concepts on quality of service in packet networks; the advanced techniques for intra-domain and inter-domain routing; the main technologies for local, data center, metro and wide area networks; network systems architectures; the issues of internetworking across complex, multi-domain infrastructures; technologies and methodologies for traffic engineering on flow-switched and packet-switched networks; architectures and protocols for network management; reliable provisioning of communication services; service level agreement design and implementation; the problems related to the secure and reliable provisioning of communication services; the advanced topics related to multicasting.	
<b>Contenuti:</b> <b>Part I. Operation and Management</b> Introduction to Network Engineering. A standard model for Operation and Management of networks and services <b>Part II. Network Architectures</b> ATM. MPLS. Wide Area Optical Networks: SONET/SDH; WDM; OTN. Metropolitan and Access Optical Networks: MetroEthernet; PON; GPON Signalling and Software Based Control: Signalling protocols; Software Defined Networks <b>Part III. Internetworking with IP</b> Packet scheduling and Quality of Service. Intradomain Routing. Interdomain Routing <b>Part IV. Network Systems Architectures</b> Switches. Adapters. Routers. Data Center and Storage LAN <b>Part V. Network Management and Control</b> Control Plane. Element Management (SNMP). Management Architectures OSS and SNMP Programming. SLA definition and management <b>Part VI. Resiliency of Networked Infrastructures</b> Principles of Fault Tolerant Design. Design Issues. Management Issues <b>Part VII. Case Studies</b> GARR-X. Google. Akamai	
<b>Codice:</b>	<b>Semestre:</b>
<b>Prerequisiti:</b> Reti di Calcolatori I	
<b>Metodo didattico:</b> lectures, lab-work, seminars	
<b>Materiale didattico:</b> <i>Text books:</i> - Larry Peterson & Bruce Davie, Computer Networks, A system approach. Fifth Edition, Morgan Kauffman – ISBN : 978-0123850591 - Mani Subramanian, Network Management – Principles and Practices, Pearson, ISBN 978-81-317-3404-9 <i>Selected readings from:</i> - Dimitrios Serpanos & Tilman Wolf, Architecture of Network Systems, Morgan Kauffman, ISBN: 978-0-12-374494-4 <i>Course slides, reading papers</i>	
<b>Modalità di esame:</b> The final mark will be assigned on the basis of the following parameters: class work and attendance: 20%; project work: 40 %; final exam: 40 %	

<b>Insegnamento: Data Mining</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<p><b>Obiettivi formativi:</b> Obiettivo del corso è illustrare le principali tecniche di data mining e le metodologie di gestione e sviluppo di un processo di data mining, dalla preparazione dei dati alla valutazione dei risultati, e di sviluppare competenze pratiche nella generazione, nell'analisi e interpretazione dei risultati mediante esercitazioni pratiche svolte con tool commerciali e/o open source.</p>	
<p><b>Contenuti:</b>  Knowledge Representation: Trees, Rules, Clusters.  Basic Methods: Statistical Modeling, Linear Models, Clustering  Performance Estimation: CV, LOO, Cost-sensitive classification, ROC curves  Advanced Data Mining: Decision Trees; Support Vector Machines, MLP, Bayesian Network, Hierarchical Clustering, EM, Semisupervised Learning  Data Transformation: Attribute selection, PCA, sampling, Cleansing  Ensemble Learning: Bagging, Boosting, Stacking, ECOC  Deep Learning.  Applications.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Materiale didattico:</b> Libro di testo: Data mining: practical machine learning tools and techniques.— 3rd ed. / Ian H. Witten, Frank Eibe, Mark A. Hall. —The Morgan Kaufmann	
<b>Modalità di esame:</b> Elaborato, Prova Orale	

<b>Insegnamento: Impianti di Elaborazione</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 52	<b>Ore di esercitazione:</b> 20
<b>Anno di corso:</b> II	
<p><b>Obiettivi formativi:</b> L'obiettivo del corso di Impianti di Elaborazione è di fornire gli elementi conoscitivi utili alla scelta, al dimensionamento, alla valutazione e alla gestione di un moderno Impianto di Elaborazione.</p> <p>Al fine di raggiungere efficacemente i suoi obiettivi, il corso affronta, da una prospettiva squisitamente ingegneristica, gli argomenti relativi alle modalità di acquisizione, agli aspetti architettonici, alla modellazione per l'analisi delle prestazioni e dell'affidabilità, al dimensionamento e alle tecniche di analisi sperimentale di un impianto di Elaborazione.</p>	
<p><b>Contenuti:</b></p> <p><b>Concetti Introduttivi</b> Aspetti caratterizzanti di un impianto di elaborazione. La preparazione di un sito. Modalità di acquisizione di un impianto di elaborazione. La scrittura di un capitolo tecnico. Le preparazioni di una proposta tecnica. Il collaudo di un impianto. Aspetti caratterizzanti la gestione di un impianto di elaborazione. Business Continuity plan e Tecniche di Disaster Recovery. Tecniche di Backup e di Restore.</p> <p><b>Aspetti Architettonici di un impianto di elaborazione</b> Cenni su architetture parallele. La virtualizzazione: process virtual machine e system Virtual Machine. Gli Hypervisor. Solaris Zone. Xen, VMware. Il Cloud Computing. Allocazione Scheduling delle risorse in un'infrastruttura Cloud. Modalità di Pricing di un Cloud. Data Replication and Placement. Strategie per l'efficienza energetica della server-farm. Presentazioni di alcune infrastrutture per il Cloud Computing.</p> <p><b>Tecniche di modellazione per il dimensionamento di un impianto</b>  Modelli per la misura delle performance e modelli per la misura di attributi di affidabilità. Modelli Analitici e Modelli simulativi. Teoria delle code. Analisi di cosa singola. Processi di nascita e morte. Reti di Code. Legge di Little. Interactive response time. Analisi dei bottleneck. Algoritmo di convoluzione. Sistemi Timesharing. Casi di studio: Architettura SOA.</p> <p><b>Strumenti e Tecniche per l'Analisi Sperimentale delle prestazioni.</b> Tipologie di Workload. Application Benchmarks. Criteri per la selezione del workload. Caratterizzazione del Workload. Classificazione dei monitor. Monitor hardware. Monitor Software. Monitor Distribuiti. Capacity Planning e Benchmark. Tecniche per l'analisi dei dati sperimentali. Intervalli di confidenza. Modelli di regressione. Experimental Design. Simple Design. Full Factorial Design. Fractional Factorial Design. Esperimenti ad un fattore. Esperimenti "two-factor full factorial".</p> <p><b>Modelli e Tecniche per la valutazione dell'affidabilità.</b> Definizione di Dependability: Availability, Reliability, Safety. Misure di Dependability. Fault, Error e Failure. Tecniche di Fault avoidance e tecniche di Fault Tolerance. Duplication. N-Modular Redundancy. Hardware redundancy. Modelli per la valutazione dell'affidabilità: Reliability Block Diagrams, Fault Trees. Modellazione e valutazione di un TMR. Casi di studio sulla modellazione combinatoriale. Modelli di availability e reliability tramite catene di Markov. Casi di studio reali. Definizione di Hazard e di Rischio. La FMEA. Standard per la Safety: IEC61508, DO178B, CENELC EN 50128. Failure Data Analysis. Data Collection. Data Filtering. Coalescence. Failure Analysis. Real World Case Studies. Cenni sulle strategie di software Fault Tolerance (N-Version e Recovery Block)</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b>	
<b>Materiale didattico:</b> Raj Jain, The art of Computer systems Performance Analysis, Wiley Dispense didattiche consultabili dal sito docenti	
<b>Modalità di esame:</b> Modalità di accertamento orale con discussione dei progetti preparati durante il corso	

<b>Insegnamento: Progettazione e sviluppo dei sistemi software</b>	
<b>CFU: 9</b>	<b>SSD: ING-INF/05</b>
<b>Ore di lezione: 52</b>	<b>Ore di esercitazione: 20</b>
<b>Anno di corso: I</b>	
<b>Obiettivi formativi:</b>	
<p>Il corso di Progettazione e sviluppo dei sistemi software ha l'obiettivo di fornire conoscenze e competenze avanzate relative alla progettazione, modellazione, documentazione e sviluppo dei sistemi software. A tale scopo, a partire da un approfondimento sui principali modelli di architetture software impiegabili nella realizzazione di sistemi reali, si affronta il problema della modellazione e documentazione di tali architetture secondo diverse viste e prospettive, impiegando diversi linguaggi e standard di documentazione. Viene inoltre presentata una metodologia di progettazione object-oriented basata su UML di tipo Model-Driven ed il suo utilizzo per la realizzazione di applicazioni aderenti a diversi modelli di architetture software. Il corso approfondisce inoltre la progettazione di dettaglio di architetture software sequenziali e concorrenti attraverso l'impiego di Design Patterns, Framework object-oriented e tecnologie Java.</p>	
<b>Contenuti:</b>	
<p><b>Architetture Software: concetti fondamentali.</b> Definizioni di Architettura Software. Componenti e Connettori di una architettura. Modelli, Viste e Notazioni per la specifica di una architettura: Vista Strutturale, Dinamica, di Distribuzione e Vista dei Componenti.</p> <p><b>Documentare le Architetture Software.</b> Stili e modelli per documentare le diverse viste di una architettura software: stili di documentazione per la vista modulare, dei componenti, la vista di distribuzione, il comportamento, le interfacce dei componenti di una architettura.</p> <p><b>Stili Architetturali.</b> Stili influenzati dal linguaggio (procedurale e object-oriented), Stile a Livelli (Macchine Virtuali e Client-Server), Stile Data-Flow (Batch e Pipe-and-Filter), Stile basato su Interprete (Codice Mobile), Stile ad Invocazione Implicita (Publish-Subscribe e Event-Based), Peer-to-Peer, Stile C2 (Componenti e Connettori).</p> <p><b>Pattern Architetturali.</b> <i>Pattern Strutturali:</i> Pattern di Controllo Centralizzato, Pattern di Controllo Distribuito, Pattern di controllo gerarchico, Pattern a Livelli di Astrazione, Pattern multipli client/multipli servizi, Pattern multipli client/un servizio, Pattern con client multi-tier / un servizio. <i>Pattern di Comunicazione architetturale:</i> Asynchronous Message Communication Pattern, Asynchronous Message Communication with Callback, Bidirectional Asynchronous Message Communication, Broadcast Pattern, Broker Forwarding Pattern, Broker Handle Pattern, Call/Return, Negotiation Pattern, Service Discovery Pattern, Service Registration, Subscription/Notification Pattern, Synchronous Message Communication with Reply, Synchronous Message Communication without Reply.</p> <p><b>Qualità delle Architetture Software.</b> Attributi di qualità osservabili a run-time (performance, security, availability, functionality, usability), Attributi non osservabili a run-time (modifiability, portability, reusability, integrability, e testability), attributi di qualità intrinseca dell'architettura (conceptual integrity, correctness, and completeness, buildability).</p> <p><b>Metodologia COMET:</b> Una metodologia di modellazione e progettazione software basata sul linguaggio di modellazione UML 2. La modellazione dei Requisiti, Costruzione del Modello di Analisi, Progettazione dell'Architettura, Costruzione incrementale del Software, Integrazione Incrementale e Testing di sistema. La progettazione di architetture object-oriented, client-server, service-oriented, component-based ed event-based usando l'approccio proposto da COMET.</p> <p><b>Sviluppo di architetture software con tecnologie Java.</b> <i>Programmazione sequenziale in Java:</i> Generalità sul linguaggio Java, Classi, Metodi, attributi, Tipi scalari e Costrutti del linguaggio, Classi I/O, Array e Matrici, Stringhe, Classi Contenitore, Gestione delle eccezioni, Ereditarietà, Polimorfismo, Classi Astratte, Interfacce, Implementazione, GUI in Java. <i>Programmazione concorrente in Java:</i> Richiami su Processi, Thread, cooperazione, competizione, interferenza. Threads in Java: creazione, context switch, stati, problemi di concorrenza con i thread. Mutua esclusione e regioni critiche: Classi, metodi e blocchi synchronized. Implementazione di Monitor. Cooperazione fra thread: i metodi wait( ), notify( ), notifyAll( ), il problema produttore/consumatore. Il package java.util.concurrent in Java 1.5: semafori e barriere. Il Framework Hibernate.</p>	

**Design Patterns.** I patterns creazionali , strutturali, comportamentali. Singleton, Composite, Decorator, Proxy, Adapter, Facade, Observer, State e Strategy. Implementazioni in Java.

**Testing di sistema e di Integrazione di Architetture Software:** Automazione del Testing con il framework JUnit.

**Codice:**

**Semestre: II**

**Prerequisiti / Propedeuticità:**

**Metodo didattico:** lezioni frontali, esercitazioni

**Materiale didattico:** trasparenze dalle lezioni, libri di testo:

- N. Taylor, N. Medvidovic, E. Dashofy, "Software Architecture-foundations, theory and practice", Wiley 2010.

- Hassan Gomaa, «Software Modeling and Design», Cambridge Univ. Press, 2011

- Clements, Bachmann, Bass, etc. «Documenting Software Architectures- Views and Beyond», Addison Wesley, Second Ed., 2010.

- S.J. Metsker, Design Patterns in Java, Pearson, 2003.

**Modalità di esame:** La verifica prevede un progetto e una prova orale.



<b>Insegnamento: Protocolli per reti mobili</b>	
<b>CFU: 6</b>	<b>SSD: ING-INF/05</b>
<b>Ore di lezione: 38</b>	<b>Ore di esercitazione: 10</b>
<b>Anno di corso: II</b>	
<b>Obiettivi formativi:</b>	
<p>Scopo del corso è impartire una conoscenza approfondita delle architetture e dei protocolli per la gestione della mobilità. Tale scopo è perseguito attraverso l'analisi delle problematiche relative alle reti wireless ed ai sistemi mobili e la presentazione delle più recenti soluzioni proposte dai principali enti internazionali di standardizzazione. Il corso è focalizzato principalmente sulle problematiche relative all'accesso al mezzo e all'instradamento nelle reti wireless. Gli obiettivi formativi principali sono: la conoscenza dei principali algoritmi distribuiti per l'accesso al mezzo wireless; l'acquisizione delle principali metodologie per l'analisi delle prestazioni delle tecniche di accesso wireless; la conoscenza delle problematiche di sicurezza nelle reti wireless; la conoscenza delle recenti architetture di backbone wireless; la comprensione delle problematiche legate al supporto della mobilità; la conoscenza dei protocolli per il supporto della mobilità e del multihoming; la capacità di utilizzare strumenti per il monitoraggio, la gestione e la configurazione di reti wireless.</p>	
<b>Contenuti:</b>	
<p>Reti WLAN: architetture, definizione dei componenti, procedure di gestione e controllo. Reti WLAN: cenni sul livello fisico (IEEE 802.11a/b/g standards). Livello MAC: DCF e PCF. La sicurezza nelle reti WLAN: meccanismi per l'autenticazione e algoritmi di cifratura (WEP, TKIP, CCMP). Estensioni per il supporto della Qualità del Servizio: lo standard IEEE 802.11e e le funzioni di accesso al mezzo EDCA e HCCA. Metodologie per la valutazione delle prestazioni delle reti WLAN basate su IEEE 802.11. Evoluzione delle reti WLAN: l'emendamento 802.11n. Reti wireless ad-hoc: scenari applicativi, problematiche, protocolli di routing reattivi e proattivi. Reti wireless mesh: scenari applicativi, procedure di gestione e controllo, protocolli per la selezione dei percorsi, interoperabilità con altri segmenti di rete LAN. Broadband wireless: cenni su WiMax. L'impatto della mobilità sui protocolli di livello trasporto. Limiti del protocollo IP nel supporto alla mobilità. Mobile IP: architettura, funzionamento, estensioni per superare l'ingress filtering e l'attraversamento dei NAT. IPsec: protocolli AH e ESP. La tecnica Diffie-Hellman per lo scambio di chiavi in sicurezza. La gestione della mobilità e del multihoming: il protocollo HIP. La sicurezza in HIP. Applicazioni di HIP ed integrazione con altri protocolli. Strumenti e metodi per la simulazione di reti mobili.</p>	
<b>Codice:</b>	<b>Semestre: II</b>
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> Slides del corso, libro di testo: Stefano Avallone, "Protocolli per Reti Mobili". McGraw-Hill Italia. ISBN: 978-88-386-7414-3	
<b>Modalità di esame:</b> Colloquio orale sugli argomenti del corso.	

<b>Insegnamento: Ricerca operativa</b>	
<b>CFU:</b> 9	<b>SSD:</b> MAT/09
<b>Ore di lezione:</b> 60	<b>Ore di esercitazione:</b> 12
<b>Anno di corso:</b> I	
<p><b>Obiettivi formativi:</b>  Il corso ha l'obiettivo di fornire gli strumenti metodologici di base per analizzare e risolvere problemi di ottimizzazione multidimensionale lineare e non lineare (continua e intera) attraverso modelli di programmazione matematica. In particolare a fine corso lo studente sarà in grado di formulare e risolvere problemi di programmazione lineare, conoscerà i problemi e gli algoritmi fondamentali di ottimizzazione su rete e gli elementi di base di ottimizzazione intera.</p>	
<p><b>Contenuti:</b></p> <p><b>- Introduzione alla Ricerca Operativa.</b>  Decision problem solving, ottimizzazione e problemi di programmazione matematica.</p> <p><b>- Problemi di ottimizzazione continua.</b>  Cenni di ottimizzazione non lineare monodimensionale e multidimensionale (non vincolata e vincolata).</p> <p><b>- Ottimizzazione lineare continua (PL).</b>  Elementi di algebra lineare e geometria poliedrale; formulazione di problemi P.L.; rappresentazione grafica di un problema P.L.; algoritmo del Simplexso standard; struttura algebrica della PL e algoritmo del simplexso revisionato; analisi post-ottimale e analisi parametrica; teoria della dualità.</p> <p><b>- Ottimizzazione lineare intera (PLI).</b>  Introduzione alla formulazione di problemi di ottimizzazione lineare intera e binaria; principali problemi applicativi ed elementi fondamentali degli algoritmi risolutivi (cutting plane e branch and bound).</p> <p><b>- Teoria dei grafi.</b>  Elementi di teoria dei grafi e proprietà topologiche; algoritmi di visita; strutture dati di un grafo.</p> <p><b>- Ottimizzazione su rete.</b>  Modellazione di problemi di ottimizzazione su rete tramite PL e PLI; introduzione ai problemi ed ai modelli base di percorso, flusso e albero minimo ed algoritmi risolutivi; algoritmi di Dantzig, Dijkstra, Floyd, simplexso su rete; Ford e Fulkerson, Sollin e Kruskal.</p> <p><b>- Tecniche reticolari di programmazione e controllo.</b>  PERT (Program Evaluation and Review Technique).</p>	
<b>Codice:</b> 00147	<b>Semestre:</b> II
<b>Propedeuticità:</b> Algoritmi e strutture dati	
<b>Metodo didattico:</b> Il corso si articolerà attraverso lezioni frontali di tipo teorico, esercitazioni di tipo numerico e di introduzione all'uso di software di ottimizzazione	
<p><b>Materiale didattico:</b>  A. Sforza, Modelli e Metodi della Ricerca Operativa, 2a ed., ESI, Napoli.  F. S. Hillier, G. J. Lieberman, Ricerca operativa - Fondamenti, 9/ed., McGraw-Hill.  Slides e dispense integrative fornite dal docente.</p>	
<b>Modalità di esame:</b> Prova scritta e colloquio orale.	

<b>Insegnamento: Secure systems design</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 48	<b>Ore di esercitazione:</b>
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b>	
<p>Obiettivo del corso è fornire gli elementi metodologici di base, le conoscenze tecniche e gli strumenti per progettare sistemi di elaborazione sicuri. In particolare, il corso di Secure System Design (Progettazione di Sistemi Sicuri) mira a formare specialisti in grado di comprendere le principali problematiche di progettazione, sviluppo e gestione di sistemi sicuri con una visione organica dei meccanismi e delle procedure di sicurezza da implementare a tutti i livelli del sistema.</p>	
<b>Contenuti:</b>	
<p>Concetti di base e fondamenti di crittografia: requisiti di un sistema sicuro; principi base della crittografia; crittografia a chiave simmetrica e asimmetrica; esempi dei più diffusi algoritmi; firma digitale e infrastruttura a chiave pubblica (PKI); algoritmi per la gestione delle chiavi.</p> <p>Controllo degli accessi: autenticazione, autorizzazione e auditing; modelli e standard per il controllo degli accessi (MAC, DAC, RBAC); framework e protocolli per il controllo degli accessi (SAML, XACML, OAuth); identità federate e specifiche WS-* per la sicurezza; problematiche di privacy dei dati.</p> <p>Progettazione di sistemi sicuri: problematiche di sicurezza nei vari livelli di un'infrastruttura complessa (sicurezza di rete, sicurezza dei sistemi operativi, sicurezza nei database, sicurezza nelle applicazioni, tecniche di programmazione sicura, ....); minacce, contromisure e meccanismi di sicurezza; sicurezza nel Cloud.</p> <p>Progettazione di sistemi sicuri embedded: sicurezza nei dispositivi mobili e dedicati (smartphone, sensori, FPGA,...), tecniche di riconfigurazione per la sicurezza.</p> <p>Gestione e valutazione della sicurezza: Business Continuity Plan e Disaster Recovery, Risk management, Security Evaluation Criteria, standard per la sicurezza.</p>	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> il corso è costituito da 24 lezioni da due ore ciascuna, le lezioni sono di carattere sia teorico che pratico	
<b>Materiale didattico:</b> è consigliato l'uso di libri di testo da usare a supporto delle lezioni. Inoltre, a tutti gli studenti, verranno fornite le slide del docente ed altro materiale disponibile liberamente sul web.	
<b>Modalità di esame:</b> l'esame consiste in una prova orale in cui gli studenti dovranno discutere un elaborato teorico e/o pratico concordato precedentemente con il docente e dimostrare la conoscenza dei contenuti del corso	

<b>Insegnamento: Sistemi distribuiti</b>	
<b>CFU: 6</b>	<b>SSD: ING-INF/05</b>
<b>Ore di lezione: 36</b>	<b>Ore di esercitazione: 12</b>
<b>Anno di corso: II</b>	
<b>Obiettivi formativi:</b>	
<p>Il corso ha l'obiettivo di illustrare:</p> <ul style="list-style-type: none"> <li>- i problemi avanzati tipici dei sistemi software distribuiti, relativi alla comunicazione, alla cooperazione e alla competizione tra processi, e i modelli e gli algoritmi per la loro risoluzione;</li> <li>- i principali aspetti che caratterizzano i sistemi operativi quando evolvono da un ambiente monolitico a un ambiente distribuito.</li> </ul>	
<b>Contenuti:</b>	
<p><u>Caratterizzazione di un sistema distribuito.</u> Sistemi distribuiti sincroni e asincroni, condivisione delle risorse, concorrenza, scalabilità, tolleranza ai guasti.</p> <p><u>Sistemi operativi distribuiti.</u> S. O. di rete e S.O. distribuiti. Elementi caratterizzanti un S. O. distribuito: trasparenza, flessibilità, affidabilità, prestazioni, real-timeliness.</p> <p><u>Sincronizzazione.</u> Sincronizzazione dei clock, tempo logico e orologi logici, algoritmi di coordinazione distribuita, mutua esclusione, algoritmi di elezione, problemi di consenso in presenza di guasti, controllo di concorrenza ottimistico, timestamp ordering, stallo.</p> <p><u>Comunicazione.</u> Comunicazioni di gruppo affidabili e non affidabili. Caso di studio: comunicazione di gruppo in ambiente Unix.</p> <p><u>Consistenza.</u> Consistenza dei dati, serializzabilità, transazioni, two-phase lock, effetto domino, memoria stabile, azioni atomiche multiprocesso, two-phase commit, azioni atomiche nidificate.</p> <p><u>Scheduling.</u> Thread, allocazione, scheduling nei sistemi distribuiti.</p> <p><u>File system distribuiti.</u> Architettura e proprietà dei file system distribuiti.</p>	
<b>Codice:</b>	<b>Semestre: I</b>
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> Slides del corso, libri di testo:	
<b>Modalità di esame:</b> Elaborato e prova orale.	

<b>Insegnamento: Sistemi Embedded</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 30	<b>Ore di esercitazione:</b> 18
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b>	
<p>Il corso fornisce le conoscenze metodologiche e tecnologiche per l'analisi e la sintesi dei moderni "sistemi embedded", e cioè quei sistemi informatici speciali e generali progettati per essere integrati in prodotti utilizzati in ambito industriale (avionica, meccanica, trasporti, chimica, ecc) e di largo consumo (telefonia, intrattenimento, elaborazione multimediale, etc.), vincolati, spesso, anche a soddisfare taluni requisiti di tipo real-time e prestazionali, oltre che requisiti su consumi, ingombro, affidabilità e sicurezza.</p> <p>Lo studente è avviato alla progettazione di sistemi embedded basati su architetture di tipo SoC (System on Chip), MPSoC (Multi Processor on Single Chip) e speciali (DSP, hardware dedicato) realizzate anche con tecnologie FPGA. Per la progettazione si fa ricorso a metodologie di sviluppo e ad ambienti IDE ampiamente impiegati nell'industria.</p> <p>Il corso include un'ampia parte esercitativa sviluppata utilizzando ambienti IDE industriali (composti da compilatori di linguaggi HDL - VHDL, VERILOG e System-C, debugger, simulatori e da strumenti per il mapping tecnologico) e lo sviluppo di un complesso progetto d'aula, per la realizzazione di un completo sistema embedded, organizzato in sottosistemi, ciascuno assegnato a un gruppo di lavoro.</p> <p>Il corso fa ampio riferimento agli argomenti già trattati nei corsi di Fondamenti di Informatica, Calcolatori Elettronici I, Sistemi Operativi e Architettura dei Sistemi di Elaborazione.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni frontali, esercitazioni	
<b>Materiale didattico:</b> slide dalle lezioni, libri di testo	
<b>Modalità di esame:</b> La verifica prevede una prova scritta e una prova orale.	

<b>Insegnamento: Sistemi Informativi</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 32	<b>Ore di esercitazione:</b> 16
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso si pone l'obiettivo di fornire le basi di architettura, progettazione e gestione dei moderni sistemi informativi al fine di sfruttare le potenzialità delle attuali tecnologie dell'informazione, sia come strumento al servizio degli obiettivi aziendali, sia come catalizzatore dell'innovazione organizzativa e strategica. Vengono inoltre affrontati i principi metodologici di alcune fasi del ciclo di vita di un sistema informativo, con riferimento non solo agli aspetti tecnologici, ma anche a quelli che richiedono attenzione al contesto organizzativo ed economico.	
<b>Contenuti:</b> <b>Parte Prima</b> <b>I Sistemi Informativi Aziendali:</b> Introduzione ai Sistemi Informativi Aziendali. Il modello organizzativo, funzionale ed informatico di un Sistema Informativo. Sistemi Informativi Operativi vs Sistemi Informativi Direzionali. <b>I Processi Gestionali nei Sistemi Informativi:</b> Tipologie di classificazione dei processi. Identificazione, descrizione e scomposizione dei processi. Cenni alla Modellazione ed al Workflow dei Processi. Prestazioni dei Processi Gestionali. Business Process Reengineering (BPR). <b>Le Tecnologie Informatiche alla base dei Sistemi Informativi:</b> Architettura dei moderni Sistemi Informativi. Sistemi ERP. Sistemi CRM. Sistemi Informativi basati sul Web. Architettura SOA. Integrazione e riuso dell'esistente. <b>Parte Seconda</b> <b>Ciclo di Vita dei Sistemi Informativi:</b> Pianificazione. Assessment e Benchmarking. Reingegnerizzazione e Studio di Fattibilità. Progettazione, Realizzazione e Manutenzione. Gestione e Conduzione. Cenni al Project Management. <b>Aspetti Normativi dei Sistemi Informativi:</b> Procedure di acquisizione di Sistemi Informativi (gare di appalto, contratti, outsourcing). Disciplina Tecnica e Procedure per la selezione del Fornitore. <b>Esempi di Sistemi Informativi:</b> Sistemi Informativi per la Logistica e la Produzione. Sistemi Informativi per i Trasporti. Sistemi informativi Sanitari. Sistemi Informativi per la Pubblica Amministrazione. Sistemi Informativi Territoriali. Sistemi Informativi a supporto della ricerca (CRIS).	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> Slides del corso, libri di testo: Bracchi, Francalanci, Motta: Sistemi informativi per l'industria digitale. Mc Graw Hill 2010.	
<b>Modalità di esame:</b> Elaborato e prova orale	

<b>Insegnamento: Trasmissione Numerica</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/03
<b>Ore di lezione:</b> 33	<b>Ore di esercitazione:</b> 15
<b>Anno di corso:</b> I	
<b>Obiettivi formativi:</b> Acquisire familiarità con le tecniche di modulazione analogica e con quelle relative alla trasmissione numerica dell'informazione su canale gaussiano.	
<b>Contenuti:</b> Rappresentazione di segnali e processi aleatori passabanda. Caratterizzazione del rumore, rumore bianco. Schema canonico di un sistema di comunicazione analogico. Elementi di modulazione analogica. Schema canonico di un sistema di comunicazione numerico. Cenni sulla codifica di sorgente e di canale. Tecniche di segnalazione numerica su canale additivo gaussiano bianco in banda base e in banda traslata. Segnalazioni ASK, PSK, QAM, FSK. Trasmissione su canale gaussiano a banda limitata.	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni, laboratorio	
<b>Materiale didattico:</b> Slides del corso, libri di testo:	
<b>Modalità di esame:</b> prova scritta, colloquio	

## Attività formative disponibili per la scelta autonoma dello studente

<b>Insegnamento: Advanced Computer Architecture and GPU Programming</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso fornisce conoscenze avanzate sulle architetture degli odierni calcolatori, con particolare riferimento alle diverse forme di parallelismo esposte alle applicazioni. Il corso pertanto approfondisce concetti legati alla struttura interna dei processori superscalari, per poi estendere la trattazione ai calcolatori multi-core e multi-processore. Un'ampia parte del programma è rivolta alla presentazione delle architetture di calcolo eterogenee, in particolare basate su GPU, un paradigma ormai assestato per lo sviluppo di applicazioni parallele ad alte prestazioni. A riguardo, oltre agli aspetti architettonici, il corso dedica diverse lezioni teoriche e la maggior parte della sezione esercitativa alla presentazione di modelli di programmazione per architetture GPU, in particolare CUDA ed OpenCL. Uno degli obiettivi del corso è infatti trasferire allo studente le competenze pratiche necessarie allo sviluppo di applicazioni parallele ed all'analisi dei relativi aspetti prestazionali. Allo scopo, il corso prevede anche la presentazione dettagliata di casi di studio reali, parte dei quali sviluppati in maniera interattiva con gli studenti	
<b>Contenuti:</b> Processori superscalari Architetture parallele: organizzazione della memoria, interconnessioni Piattaforme di calcolo eterogenee: GPU, GPGPU Modelli e framework per la programmazione: CUDA, OpenCL Compilatori per piattaforme di calcolo parallele Casi di studio. Esercitazioni.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Calcolatori Elettronici II	
<b>Metodo didattico:</b> Lezioni frontali, Esercitazioni guidate, Sviluppo di un elaborato da discutere in sede d'esame	
<b>Materiale didattico:</b> Appunti forniti dal docente Materiale esercitativo Libri di testo: <ul style="list-style-type: none"><li>• J. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 5th Edition, Morgan Kaufmann 2012 (Appendix C and Chapter 3)</li><li>• D. J. Sorin, M. D. Hill, and D. A. Wood, A Primer on Memory Consistency and Cache Coherence, Morgan Claypool 2011</li><li>• N. E. Jerger and L.-S. Peh, On-Chip Networks, Morgan Claypool 2009</li><li>• NVIDIA, CUDA C Programming Guide, v. 7.5, available online, NVIDIA 2015</li></ul>	
<b>Modalità di esame:</b> Discussione di un elaborato sviluppato dallo studente Valutazione orale	



<b>Insegnamento: Analisi e Prestazioni di Internet</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> La rete Internet riveste sempre più importanza nella vita e nelle attività di tutti i giorni. Per il suo corretto dimensionamento e funzionamento e per poterne garantire proprietà quali affidabilità, requisiti prestazionali, sicurezza etc., è necessaria una profonda conoscenza e comprensione dei meccanismi base e dei principali elementi caratterizzanti di una rete: il traffico, le topologie, i servizi, le applicazioni, etc. L'obiettivo del corso di <b>Analisi e Prestazioni di Internet</b> è quello di fornire gli elementi e gli strumenti utili alla analisi ed alla valutazione delle prestazioni di una moderna rete Internet e di presentare architetture, tecnologie e protocolli per migliorare le sue prestazioni. Il corso presenta i contenuti adottando un approccio ingegneristico ed empirico e fonde lezioni teoriche, lezioni pratiche ed esercitazioni. Esso presenta gli aspetti principali e le motivazioni alla base dell'analisi e della valutazione prestazionale di una rete per poi approfondire gli aspetti metodologici e pratici legati all'analisi prestazionale di rete, con specifico riferimento alla rete Internet.	
<b>Contenuti:</b> <b>Introduzione e Concetti di Base:</b> Contestualizzazione didattico/scientifica del Corso, Terminologia di Base, Inquadramento degli aspetti principali dell'analisi e delle prestazioni di Internet e motivazioni, Risoluzione dei Problemi di Rete, Requisiti di Prestazioni delle Reti e delle Applicazioni in Rete. <b>Fondamenti di Analisi e Monitoraggio delle Prestazioni di Internet:</b> Background Analitico (probabilità, statistica, rappresentazione dei dati, forecasting, grafi, etc.), Metriche per l'Analisi ed il Monitoraggio di Internet, Modelli per l'analisi delle caratteristiche e delle prestazioni di Internet, Approcci al Monitoraggio di Internet (attivo, passivo, ibrido, etc.), Protocolli per il controllo ed il monitoraggio di Internet, "Practical Issues" nell'Analisi e nel Monitoraggio di Internet, Standardizzazione delle procedure, delle metriche, dei protocolli, e delle piattaforme per il Monitoraggio di Internet (IETF, ITU.T, etc.), Attività di Monitoraggio di Internet operate da Authority: FCC e AGCOM, Network science e Web science. <b>Metodologie e Tecniche di Analisi e Monitoraggio delle Prestazioni di Internet:</b> Metodologie e Tecniche per l'analisi del Traffico di Internet: workload di rete, caratterizzazione e modelling statistico, identificazione e classificazione, modelli di generazione, etc.; il monitoraggio della Banda: capacità e banda disponibile in reti wired e wireless; Traceroute e il Monitoraggio di percorsi di Rete e delle Topologie di Rete Internet (livello IP, livello Router, livello PoP, livello AS); il Monitoraggio dei parametri di Qualità di Rete (ad es., throughput, delay, jitter, RTT's, etc.); il Monitoraggio dei parametri di Qualità delle Applicazioni in Rete (metriche e metodologie per la valutazione della Quality of Experience), Monitoraggio e analysis attraverso SNMP/RMON, NetFlow/IPFIX. Simulazione ed Emulazione di rete. <b>Architetture, Tecnologie e Protocolli per migliorare le Prestazioni di Internet:</b> Varianti del protocollo TCP, SCTP, DCCP, SDN, Middlebox, Load balancing, <b>Applicazioni dell'Analisi di Internet:</b> Analisi e Monitoraggio di DNS, Web, P2P, Streaming, etc.; Analisi e Performance di TCP; Valutazione di SLA; Monitoraggio di Security e Outages, Neutralità di Rete e Censura, Routing, Online Social Network, Online Games; Analisi, Monitoraggio e Prestazioni di SDN, Cloud e Data Center; Visualizzazione dei dati di monitoraggio;.	
<b>Casi di Studio:</b> Piattaforme sperimentali per l'analisi e il monitoraggio di Internet, Piattaforme di monitoraggio su larga scala, Analisi delle Prestazioni di Applicazioni Internet in esercizio.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Reti di Calcolatori	
<b>Metodo didattico:</b> Lezioni Teoriche con supporto Multimediale, Lezioni Pratiche, Esercitazioni.	
<b>Materiale didattico:</b> Slide e Dispense del Docente, Librop di testo Consigliato: Internet Measurement: Infrastructure, traffic & applications, Mark Crovella, Balachander Krishnamurty, Wiley.	
<b>Modalità di esame:</b> Homework, Progetto, Colloquio Orale.	

<b>Insegnamento: Architettura dei sistemi integrati</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/01
<b>Ore di lezione:</b> 58	<b>Ore di esercitazione:</b> 14
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Capacità di progettare ed analizzare a livello architetturale, circuitale e fisico circuiti e sistemi digitali VLSI. Conoscenza dei linguaggi per la descrizione dell'hardware. Capacità di utilizzare sistemi di sviluppo per la progettazione assistita al calcolatore di sistemi VLSI. Conoscenza delle tecniche di testing dei sistemi digitali.	
<b>Contenuti:</b> Classificazione dei sistemi integrati: full-custom, basati su celle standard e programmabili. Metodologie di progetto di sistemi integrati. Tecniche di sintesi e di place and-route automatiche. Tecniche di simulazione switch-level. Livelli di interconnessione e parametri parassiti. Ritardi introdotti dalle interconnessioni. Elmore delay. Static timing analysis. Progetto di sistemi combinatori. Progetto e temporizzazione di sistemi sequenziali. Pipelining. Generazione e distribuzione del clock. PLL, DLL. Linguaggi per la descrizione dell'hardware. Il VHDL per la descrizione e la sintesi di sistemi integrati. Circuiti aritmetici: Addizionatori, Unità logico-aritmetiche, Moltiplicatori. Testing dei sistemi integrati CMOS. Tecniche di self-testing. Valutazione della dissipazione di potenza nei sistemi VLSI. Tecniche per la riduzione della dissipazione di potenza.	
<b>Codice:</b> 01577	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Conoscenza di base dei sistemi digitali, delle principali caratteristiche di dispositivi MOS e delle logiche CMOS	
<b>Metodo didattico:</b> Lezioni, esercitazioni al calcolatore, seminari.	
<b>Materiale didattico:</b> Appunti del corso disponibili sul sito docente. Testi di riferimento: Weste, Harris: "CMOS VLSI Design – circuit and systems perspective" Pearson – Addison Wesley; Rabaey "Circuiti Integrati Digitali, l'ottica del progettista", II Edizione, Pearson - Prentice Hall	
<b>Modalità di esame:</b> Colloquio, discussione dell'elaborato sviluppato durante le esercitazioni.	

<b>Insegnamento: Automazione dei processi di business</b>	
<b>CFU:</b> 3	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 18	<b>Ore di esercitazione:</b> 6
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Obiettivo del corso è fornire agli studenti i principali concetti relativi al workflow management. Il focus del corso è sulla definizione, rappresentazione e codifica del workflow, mediante l'utilizzo di linguaggi (Business Process Modeling Notation, BPMN; Business Process for Execution Language, BPEL) e design patterns.	
<b>Contenuti:</b> La gestione dei processi di business (Business Process Management, BPM) è una disciplina che integra aspetti relativi alla gestione di impresa e l'information technology. Studia la definizione, l'ottimizzazione, l'integrazione e il monitoraggio dei processi aziendali ed ha acquistato negli ultimi anni una rilevanza sempre maggiore poichè fornisce validi strumenti per rendere efficienti ed efficaci i processi aziendali aumentando la produttività e riducendo i costi. In questo contesto il principale obiettivo del workflow management è l'automazione dei processi di business, che richiede l'applicazione di modelli, metodi e linguaggi per la modellazione, l'analisi, l'implementazione e l'enactment dei flussi di lavoro (workflows). In tale contesto il corso tratterà i seguenti argomenti: - Introduzione ai sistemi di gestione del workflow e relativi standards (WfMS) con cenni ad attuali realizzazioni; - Definizione dei processi di workflow, orchestrazione e coreografia; - I linguaggi BPEL e/o BPMN per la realizzazione di processi di business; - Workflow patterns e loro realizzazione in BPEL; - Applicazione a domini reali.	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> Lezioni frontali, sviluppo di esempi	
<b>Materiale didattico:</b> Riferimenti e risorse reperibili on-line, materiale fornito dal docente	
<b>Modalità di esame:</b> Elaborato e esame orale	

<b>Insegnamento: Calcolo Parallelo</b>	
<b>CFU:</b> 6	<b>SSD:</b> MAT/08
<b>Ore di lezione:</b> 30	<b>Ore di esercitazione:</b> 18
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso intende fornire allo studente una introduzione alle metodologie ed alle problematiche alla base del disegno e progettazione di un algoritmo in un ambiente di calcolo ad alte prestazioni, nonché alle moderne architetture hardware ed al software per il calcolo parallelo e distribuito. Il corso prevede attività di laboratorio con sviluppo di software parallelo in ambiente MPI, OpenMP e ibrido su un cluster di nodi di calcolo presso il Data Center SCoPE.	
<p><b>Concetti introduttivi.</b> Architettura di un calcolatore parallelo. Classificazione di Flynn Architetture a memoria distribuita ,a memoria condivisa e ibride; multicore e GPU. La TOP500. Problematiche di base del calcolo parallelo e modelli di sviluppo di algoritmi paralleli.<b>Parametri di valutazione di un software parallelo.</b> Speed-up efficienza ed overhead di un algoritmo parallelo. Legge di Amdahl, legge di Gustafson,</p> <p><b>MPI – Message Passing Interface.</b> Metodologie e criteri per lo sviluppo di algoritmi paralleli su calcolatori MIMD-DM. Il modello message-passing: caratteristiche e funzioni di base. La Libreria MPI: caratteristiche, principali routine per la definizione dell’ambiente, per la gestione dei processi, per le comunicazioni puntuali e collettive, per le operazioni collettive. Definizione di tipi MPI. Griglie di processori.</p> <p><b>OpenMP –</b> Caratteristiche di un sistema SharedMemory(multicore).Modello di programmazione fork-join. Direttive. Costrutti di controllo. Clausole.Runtime library routines. Gestione della memoria :data race e false sharing.</p> <p><b>Algoritmo per la somma di n numeri.</b> Decomposizione del problema. Algoritmo di somma in ambiente MIMD Shared Memory(multicore). Algoritmo in ambiente MIMD Distributed Memory .Complessità di tempo, speed-up, efficienza. Algoritmi paralleli per il calcolo di un integrale.</p> <p><b>Algoritmo per il prodotto matrice-vettore.</b> Algoritmi a blocchi, le strategie di distribuzione dei dati.Parallelizzazione in un ambiente Distributed Memory .Parallelizzaione in un ambiente shared-memory.</p> <p><b>Algoritmo per il prodotto matrice-matrice.</b>Strategie di distribuzione dei dati a blocchi. Algoritmo BMR. Parallelizzazione in un ambiente Distributed Memory : distribuzione dei dati e comunicazioni.</p> <p><b>Modello di programmazione ibrida MPI-OpenMP.</b> Motivazioni , vantaggi e svantaggi della programmazione ibrida. Gestione dei nodi. Esempio di compilazione ed esecuzione.</p> <p><b>Programmazione delle GPU. Le GPU della Nvidia.</b> La programmazione in CUDA.Esempi.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Fondamenti o elementi di Informatica	
<b>Metodo didattico:</b> lezioni,esercitazioni	
<b>Materiale didattico:</b> Dispense del docente on line. Almerico Murli – “Lezioni di Calcolo Parallelo”, Liguori Editore.	
<b>Modalità di esame:</b> colloquio con discussione sul progetto svolto.	

<b>Insegnamento: Circuiti per DSP</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/01
<b>Ore di lezione:</b> 58	<b>Ore di esercitazione:</b> 14
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Conoscenza approfondita delle architetture dei circuiti DSP disponibili commercialmente e dell'ambiente di sviluppo per la loro programmazione. Conoscenza delle problematiche, sia teoriche che pratiche, relative alla implementazione ottimale, in tempo reale, su DSP, dei principali algoritmi di elaborazione digitale dei segnali. Realizzazione di concreti algoritmi di elaborazione dei segnali su circuiti DSP.	
<b>Contenuti:</b> Tecniche di calcolo avanzate in aritmetica a virgola fissa e mobile per la realizzazione di algoritmi di elaborazione dei segnali. Effetti derivanti dalla precisione finita dei segnali: quantizzazione dei coefficienti, prevenzione e gestione dell'overflow, tecniche di rounding. Studio dei circuiti programmabili per l'elaborazione dei segnali (DSP): sistemi di memoria multi-accesso, hardware per calcolo degli indirizzi (buffering circolare, indirizzamento bit-reversal), unità Single Instruction Multiple Data. Utilizzo delle tecniche di pipelining nei circuiti DSP. Hazards nei circuiti DSP. Architetture Very Long Instruction Word (VLIW). Tecniche di ottimizzazione del codice nei circuiti DSP con architetture VLIW: Loop Unrolling, Software Pipelining. Implementazione in tempo reale degli algoritmi di elaborazione nei circuiti DSP: interfacce seriali sincrone (buffered e multi-channel), elaborazione in streaming, elaborazione a blocchi, elaborazione in sistemi operativi real-time. Debugging ed analisi delle prestazioni in tempo reale dei circuiti DSP. Metodologie di in-system debugging.	
<b>Codice:</b>	
<b>Propedeuticità:</b> Nessuna	
<b>Prerequisiti:</b> Conoscenza di base del funzionamento dei circuiti digitali e del linguaggio C per lo svolgimento delle esercitazioni.	
<b>Metodo didattico:</b> Lezioni ed esercitazioni	
<b>Materiale didattico:</b> John G. Proakis, Dimitris G. Manolakis, " <i>Digital Signal Processing: Principles, Algorithms and Applications</i> ", 4° edition, Prentice Hall 2007 Sen M. Kuo, Woon-Seng Gan, " <i>Digital Signal Processors: Architectures, Implementations, and Applications</i> ", Prentice Hall 2005 Appunti delle lezioni	
<b>Modalità di esame:</b> Colloquio con discussione delle esercitazioni svolte	

<b>Insegnamento: Cloud e Datacenter Networking</b>	
<b>CFU:</b> 3	<b>SSD:</b> ING_INF/05
<b>Ore di lezione:</b> 24	<b>Ore di esercitazione:</b>
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Obiettivo di questo corso è di presentare le architetture di rete impiegate nel contesto dei moderni datacenter e delle infrastrutture per l'offerta di servizi di Cloud Computing. Saranno presentate sia soluzioni tradizionali che paradigmi emergenti, come quello del <i>Software Defined Networking</i> .	
<b>Contenuti:</b> <b>Architettura di un datacenter.</b> Architetture e topologie di rete per datacenter. Standard ANSI/TIA-942. Top-of-Rack design. End-of-Row design. <b>Tecnologie di networking nei datacenter.</b> Storage Networking e relative tecnologie: Fiber Channel, iSCSI, ATA over Ethernet (AoE), Fibre Channel over Ethernet (FcoE). VLAN. Protocolli di Spanning Tree e VLAN. Load Balancers nei datacenter. Il problema del TCP Incast e varianti di TCP per datacenter networks. Tecnologie di virtualizzazione e loro impiego in un datacenter. Impatto della virtualizzazione sulle infrastrutture di networking di un datacenter. Interconnessione geografica di datacenter. Ethernet over MPLS. Ethernet over IP. <b>Aspetti di networking nei sistemi di Cloud Computing.</b> Networking in sistemi di Cloud privato. Modelli di networking in OpenStack. Networking in sistemi di Cloud pubblico. Networking in Amazon EC2. Amazon Virtual Private Cloud. Amazon CloudFront content delivery service. <b>Separazione tra Data Plane e Control Plane.</b> L'approccio ForCES. Software Defined Networking. OpenFlow. OpenFlow Controllers. POX, NOX, FloodLight. Sviluppo di una applicazione di controllo per una rete OpenFlow. Uso di tecnologie SDN nel Cloud Computing. Network Function Virtualization (NFV).	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b>	
<b>Materiale didattico:</b>	
<b>Modalità di esame:</b>	

<b>Insegnamento: Economia e Organizzazione Aziendale</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING IND/35
<b>Ore di lezione:</b> 52	<b>Ore di esercitazione:</b> 20
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b>	
<p>Fornire i concetti e i modelli fondamentali, in vista delle applicazioni, relativi al comportamento degli attori economici con riferimento ai sistemi micro e macroeconomici.</p> <p>Fornire le conoscenze di base per l'analisi delle prestazioni aziendali a partire dai dati della contabilità generale d'impresa e per le decisioni di investimento.</p> <p>Fornire le conoscenze di base sulla gestione dei progetti e dei gruppi di lavoro, con particolare riferimento al settore ICT.</p>	
<b>Contenuti:</b>	
<p>PARTE I: Economia generale (micro e macro-economia)</p> <p>Definizione di economia, principio della scarsità, razionalità dell'attore economico (costi-benefici, costo opportunità, costo e beneficio marginali), definizione di microeconomia e macroeconomia. Il sistema/ciclo macroeconomico; Misurare l'attività economica: PIL e disoccupazione; Livello dei Prezzi e Inflazione; La moneta, i prezzi e la BCE. Definizione di mercato, l'economia di mercato, il mercato come meccanismo di coordinamento dell'azione collettiva, mercato e aggregazione di informazione distribuita</p> <p>curva di domanda, curva di offerta, equilibrio, efficienza, fattori che influenzano gli spostamenti delle curve di domanda e offerta, elasticità della domanda al prezzo: definizioni e determinanti, elasticità e spesa Il concetto di utilità e di utilità marginale; Curve di indifferenza e allocazione della spesa tra due beni: scelta del paniere ottimo; La domanda individuale e la domanda di mercato, Il surplus del consumatore. Funzione di produzione: breve e lungo periodo; Costi, ricavi, profitti; Classificazione dei costi: medio, marginale, fisso, variabile; Profitto contabile e profitto economico; La massimizzazione del profitto. Cenni alle forme di mercato (Concorrenza perfetta, Monopolio, Monopsonio, Concorrenza monopolistica, Oligopolio).</p> <p>PARTE II: L'analisi di bilancio e i modelli decisionali d'impresa</p> <p>Definizione di impresa ed azienda ai sensi dell'Art. 2455; cenni alle forme giuridiche di azienda (individuale e societaria); il ciclo tecnico-economico-finanziario; la creazione di valore.</p> <p>Il modello a raggiera e le ipotesi alla base della redazione del bilancio; misure di stock e di flusso; il capitale d'impresa e la sua misurazione; il reddito d'impresa.</p> <p>Lo stato patrimoniale e il conto economico: analisi delle voci e del loro significato; la nota integrativa al bilancio. La misura delle prestazioni di impresa attraverso l'analisi dei documenti di bilancio; l'analisi della struttura patrimoniale (liquidità, solvibilità e solidità); l'analisi dei flussi (redditività); la leva finanziaria. Modelli decisionali per la gestione d'impresa; l'analisi del punto di pareggio (leva operativa); tecniche e strumenti per l'analisi e la valutazione economico-finanziaria degli investimenti (il VAN, il periodo di recupero, il TIR); la misura del costo del capitale; l'analisi del rischio.</p> <p>PARTE III: La gestione dei progetti</p> <p>La definizione di progetto; la gestione per progetti; organizzare per funzioni e organizzare per processi; la gestione delle risorse; i gruppi di lavoro; il responsabile dei progetti; dinamiche organizzative dei gruppi di lavoro. Strumenti e tecniche reticolari per la gestione dei progetti; applicazioni ed esempi nel settore ICT.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Nessuna	
<b>Metodo didattico:</b> Lezioni teoriche; esercitazioni; seminari del docente su argomenti specifici; testimonianze di esperti provenienti dal mondo aziendale	

**Materiale didattico:**

Lecture e altri materiali distribuiti dal docente durante il corso e solitamente disponibili nell'area download.

Sloman J., Garrat D. (2011) Elementi di Economia, il Mulino, Bologna

Lo Storto C., Zollo G. (1999) Problemi di microeconomia, ESI, Napoli

Ulteriori testi di approfondimento saranno suggeriti dal docente.

**Modalità di esame:** Prova scritta finale; valutazione attività intermedie; colloquio orale



<b>Insegnamento: Ingegneria del Software II</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso si propone di approfondire tematiche avanzate di Ingegneria del Software quali Testing, Debugging, Manutenzione, Reverse Engineering, Qualità del software.	
<b>Contenuti:</b> Testing: Nozioni teoriche – Testing Black Box – Junit – Testing White Box – Integration Testing – User Interface Testing – Testing Automation – Testing e analisi mutazionale - Processi di Testing – Analisi statica – Debugging Manutenzione e Reverse Engineering: Manutenzione – Reengineering, refactoring, restructuring, reverse engineering, ridocumentazione – Strumenti di Build Automation – Strumenti di gestione delle versioni concorrenti del software Qualità: Modelli di qualità del software - Metriche – Strumenti di Misura del Software - Function Points - Stima dei costi	
<b>Codice:</b> 15803	<b>Semestre:</b> II
<b>Prerequisiti:</b> Ingegneria del Software	
<b>Metodo didattico:</b> Il corso si compone di lezione teoriche frontali e lezioni di esercitazione nelle quali vengono provate alcune delle tecniche e metodologie presentate rispetto a software reali.	
<b>Materiale didattico:</b> Libri: Ian Sommerville, Ingegneria del Software, Pearson Addison Wesley; Roger Pressman, Principi di Ingegneria del Software, Mc Graw Hill Materiale online: libri e articoli scientifici consigliati dal docente, diapositive presentate a lezione	
<b>Modalità di esame:</b> Per sostenere l'esame ogni studente deve preventivamente realizzare due progetti e infine sostenere una prova orale. Il primo progetto riguarda lo sviluppo di un software (ad esempio un applicazione mobile) utilizzando anche strumenti di gestione CVS e strumenti di build automation, documentando il processo di sviluppo del software realizzato. Il secondo progetto comprende attività di reverse engineering, testing e debugging su di una applicazione reale nella quale il docente ha iniettato preventivamente alcuni difetti che dovranno essere individuati e corretti dallo studente.	

<b>Insegnamento: Metodi formali</b>	
<b>CFU:</b> 3	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 18	<b>Ore di esercitazione:</b> 6
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Obiettivo del corso di metodi formali è fornire agli studenti conoscenze nell'ambito della modellazione e verifica di sistemi software e computer-based, con particolare riferimento a tecniche di verifica formale. Verranno illustrati il ruolo e l'importanza dei metodi formali nello sviluppo di sistemi complessi e verranno introdotti diversi strumenti formali utilizzati per la modellazione di sistemi e di proprietà. Infine verranno affrontati aspetti avanzati in particolare nell'ambito delle metodologie di modellazione di sistemi complessi.	
<b>Contenuti:</b> <u>Parte I:</u> Il ruolo dei metodi formali nell'ingegneria dei sistemi, i metodi formali nella certificazione dei sistemi reali, alcuni esempi tratti dal mondo reale, proprietà funzionali e non funzionali, analisi qualitativa e quantitativa. Petri nets ed estensioni tempificate per l'analisi quantitativa di proprietà temporali. <u>Parte II:</u> linguaggi formali per la specifica e l'analisi. Logiche temporali, LTL e CTL, introduzione al model checking. tecniche di sviluppo di modelli complessi, strumenti per la modellazione e la risoluzione dei modelli. <u>Parte III:</u> esercitazioni e applicazione a casi di studio.	
<b>Docente:</b>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti:</b> Nessuno	
<b>Metodo didattico:</b> lezioni, laboratorio, seminari applicativi	
<b>Materiale didattico:</b> appunti del corso, articoli scientifici	
<b>Modalità di esame:</b> discussione di un elaborato e colloquio orale sugli argomenti del corso	

<b>Insegnamento: Metodi per le Decisioni – Ricerca Operativa II</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 48	<b>Ore di esercitazione:</b> 24
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso fornisce gli elementi fondamentali per la risoluzione di problemi di ottimizzazione, con particolare riferimento alle tecniche euristiche di risoluzione. Le metodologie risolutive sono applicate per la risoluzione dei principali problemi di logistica	
<b>Contenuti:</b> Generalità sui problemi di ottimizzazione combinatoria: Metodi esatti ed euristici. Algoritmi costruttivi ed algoritmi migliorativi. Tecniche metaeuristiche: Tabu search, Simulated Annealing, Algoritmi Genetici. Cenni sulle problematiche relative all'analisi multicriteria. I problemi di localizzazione: generalità e classificazione. Problemi di copertura (set covering, maximal covering) e problemi di efficienza (plant location e p-mediana). Il progetto delle reti: albero minimo con e senza vincoli di capacità. La gestione delle scorte: definizioni fondamentali, indicatori di prestazione, strumenti di classificazione. Modelli continui: deterministici e stocastici. Il modello del lot sizing: l'algoritmo di Wagner-Whitin e le tecniche euristiche di risoluzione. La schedulazione ottimale dei processi: definizioni fondamentali, classificazione degli obiettivi e dei problemi, la notazione di Graham. Problemi statici e dinamici su macchina singola. Il sequenziamento ottimale. Problemi statici su macchine parallele. I problemi multifase: Flow Shop e Job Shop. Tecniche reticolari per la schedulazione dei progetti. I problemi di routing: i modelli del TSP e del VRP. Algoritmi costruttivi: il nearest neighbour, gli algoritmi di inserimento, l'algoritmo di saving, metodi cluster first-route second e route first-cluster second. Use pacchetti software per la risoluzione dei problemi.	
<b>Codice:</b> 30287	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> Il corso si articolerà attraverso lezioni frontali di tipo teorico, lo sviluppo di esercitazioni di tipo numerico e mediante uso di software di ottimizzazione.	
<b>Materiale didattico:</b> Libro di testo consigliato: Bruno G. (2012). Operations management: modelli e metodi per la logistica. Terza edizione, ESI, Edizioni Scientifiche Italiane, Napoli. - le slide del corso sono disponibili su <a href="http://www.federica.unina.it">www.federica.unina.it</a>	
<b>Modalità di esame:</b> Prova scritta e colloquio orale	

<b>Insegnamento: Network Security</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Obiettivo di questo corso è di presentare le principali vulnerabilità e tipologie di attacco alle reti informatiche nonché le metodologie, tecniche e strumenti per la loro identificazione e risoluzione.	
<b>Contenuti:</b>	
<ul style="list-style-type: none"> <li><b>a) Network security threats &amp; mechanisms:</b> Worm, (D)DoS, Botnet, SPAM, Phishing, Sniffer, Keylogger</li> <li><b>b) Physical layer security</b></li> <li><b>c) IP layer security:</b> Router mechanisms for security, Routing security, IPSec</li> <li><b>d) Transport and application layer security:</b> TLS/SSL, DNS</li> <li><b>e) Security of networked systems:</b> Firewall, IDS, IPS</li> <li><b>f) Scenarios of interest:</b> Long-haul network security, Data center and enterprise network security, ISP network security and traffic anomaly detection, Critical Infrastructures</li> <li><b>g) Hot topics in network security:</b> Security testing and monitoring, Wireless, Protection, Anonymity</li> </ul>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Reti di Calcolatori, Computer Networks II	
<b>Metodo didattico:</b> Lezioni frontali ed esercitazioni	
<b>Materiale didattico:</b> Libri, slide e materiale digitale	
<b>Modalità di esame:</b> Orale + elaborato	

<b>Insegnamento: Ottimizzazione</b>	
<b>CFU: 9</b>	<b>SSD: MAT/09</b>
<b>Ore di lezione: 60</b>	<b>Ore di esercitazione: 12</b>
<b>Anno di corso: II</b>	
<p><b>Obiettivi formativi:</b> Il corso ha l'obiettivo di fornire gli strumenti metodologici avanzati per definire, analizzare e risolvere problemi decisionali complessi, attraverso la formulazione di modelli di ottimizzazione matematica multidimensionale, non lineare e lineare, continua, discreta (intera e binaria) e mista intera, con particolare risalto alle problematiche di instradamento, localizzazione e progetto su rete. Il corso prevede una componente di laboratorio basata sull'uso di software per l'ottimizzazione continua e discreta. (Xpress)</p>	
<p><b>Contenuti:</b></p> <p><b>-Ottimizzazione non lineare monodimensionale.</b> Condizioni di ottimo monodimensionale; Metodi a riduzione dell'intervallo di incertezza, con e senza l'uso della derivata; Metodi a generazione di punti con uso della derivata.</p> <p><b>- Ottimizzazione non lineare multidimensionale non vincolata.</b> Metodi di gradiente; Algoritmo di discesa e salita ripida, gradiente coniugato; Analisi grafica ed esercitazioni numeriche.</p> <p><b>- Ottimizzazione non lineare e lineare multidimensionale vincolata.</b> Condizioni di ottimo nei problemi di ottimizzazione vincolata (condizioni di Kuhn-Tucker); Metodi a direzione ammissibile; Analisi grafica ed esercitazioni numeriche; Ottimizzazione lineare come caso particolare della Ottimizzazione non lineare.</p> <p><b>- Metodi avanzati di ottimizzazione lineare intera (PLI).</b> Formulazione di problemi ottimizzazione lineare intera e nocciolo convesso; Metodi avanzati di risoluzione basati su row e column generation e Branch and Bound (Branch and Cut); Tecniche di rilassamento e rilassamento lagrangiano; Metodi approssimati (euristica lagrangiana, ricerca locale); Problemi applicativi di packing, cutting e sequencing.</p> <p><b>- Problemi avanzati di instradamento, localizzazione e progetto su rete.</b> Problemi di percorso con vincoli aggiuntivi: problemi di minimo percorso vincolato (minimi percorsi attraverso specificati vertici, con finestre temporali, con risorse limitate), minimo percorso con vincoli di capacità (Quickest path e quickest flow); Problema del percorso massimo.</p> <p>- Problemi di flusso multi-commodity: modellazione e soluzione dei problemi di flusso Multi-Commodity con costi costanti e costi variabili; modellazione di problemi di massimo flusso con più sorgenti e più pozzi;</p> <p>- Problemi di covering, partitioning, location: set covering problem e maximal covering; p-centro e p-mediana, plant location, sensor placement; problemi di path location.</p> <p>- Problemi di network design: progettazione di reti affidabili.</p> <p>- Problemi di instradamento e routing: Circuito hamiltoniano e TSP; circuito euleriano; problemi di location – routing.</p> <p><b>- Software di ottimizzazione:</b> introduzione ai software di ottimizzazione (Lindo, Xpress, Cplex); modellazione e risoluzione di problemi reali di ottimizzazione lineare continua e intera.</p>	
<b>Codice: 27038</b>	<b>Semestre: I</b>
<b>Prerequisiti / Propedeuticità:</b>	
<p><b>Metodo didattico:</b> Il corso si articolerà attraverso lezioni frontali di tipo teorico, esercitazioni di tipo numerico e uso di software di ottimizzazione.</p>	
<p><b>Materiale didattico:</b> A. Sforza, Modelli e Metodi della Ricerca Operativa, 2a ed., ESI, Napoli. F. S. Hillier, G. J. Lieberman, Ricerca operativa - Fondamenti, 9/ed., McGraw-Hill. C. Guéret, C. Prins, M. Sevaux, Applications of optimization with Xpress-MP, Editions Eyrolles, Paris. IBM ILOG CPLEX V12.1 User's Manual for CPLEX. Slides e dispense integrative fornite dal docente.</p>	
<b>Modalità di esame:</b> Prova scritta e colloquio orale	

<b>Insegnamento: Reti Wireless</b>	
<b>CFU:</b> 9	<b>SSD:</b> ING-INF/03
<b>Ore di lezione:</b> 54	<b>Ore di esercitazione:</b> 18
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Conoscere le principali problematiche che caratterizzano le reti wireless, con particolare attenzione alle reti locali e metropolitane (WMAN, WLAN, WPAN, reti ad hoc e reti di sensori). Conoscere le principali tecnologie e gli standard per le reti wireless locali e cellulari.	
<b>Contenuti:</b> Generalità sulle reti e sui servizi di telecomunicazione. Convergenza nelle reti. Architetture e topologie delle reti wireless. Principali modelli per la caratterizzazione del canale wireless. Tecniche di modulazione per reti wireless. Tecniche di accesso per reti wireless. Reti mesh. Reti ad hoc. Reti di sensori. Reti per l'accesso wireless a larga banda. Principali standard per reti wireless ad estensione locale, metropolitana e geografica (WiMaX, LTE, IEEE 802.11, HIPERLAN, Bluetooth, ZigBee, WiMedia). Mobile ad-hoc networks (MANET). Cenni sul paradigma Cognitive Radio.	
<b>Prerequisiti:</b> conoscenze di base di trasmissione numerica e reti di telecomunicazioni e/o di calcolatori	
<b>Metodo didattico:</b> lezioni	
<b>Materiale didattico:</b> appunti del corso, libri di testo: K. Pahlavan, P.Krishnamurthy: Principles of Wireless Networks. Prentice Hall Communications Engineering 2002. (disponibile in biblioteca) + articoli di rassegna	
<b>Modalità di esame:</b> esposizione di un elaborato o colloquio orale	

<b>Insegnamento: Semantic Web</b>	
<b>CFU:</b> 6	<b>SSD:</b> INF/01
<b>Ore di lezione:</b> 48	<b>Ore di esercitazione:</b>
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b> Il corso si propone di fornire allo studente le basi teoriche e implementative per lo sviluppo di ontologie nell'ambito del web semantico. Nello specifico si affronterà lo standard OWL e le Logiche Descrittive sottiacenti. In tale contesto vengono studiate sia le proprietà formali delle logiche trattate (tra le quali il rapporto fra espressività e complessità) che le tecniche di implementazione (ottimizzazioni comprese). Infine, particolare enfasi verrà data all'uso delle logiche medesime per rappresentare diversi domini applicativi.	
<b>Contenuti:</b> Introduzione e motivazioni: la diffusione di tecniche di AI nelle applicazioni "di ogni giorno", e il ruolo della rappresentazione della conoscenza in questi contesti. Richiami di logica del primo ordine e complessità computazionale. Logiche descrittive: sintassi, semantica e reasoning tasks. Logiche descrittive e lo standard OWL. Espressività e complessità dei diversi frammenti logici. Strumenti di sviluppo di Ontologie (Protegé). Modellazione di domini tramite OWL (definizione di ontologie). Risultati negativi: indecidibilità di alcune logiche descrittive. Logiche a bassa complessità e il profilo OWL-EL. Meccanismi di ragionamento automatico basati su tableaux e relative tecniche di ottimizzazione. Meccanismi di ragionamento per la logica EL. Elementi di programmazione logica: il linguaggio Prolog, esempi di programmi logici, negazione come fallimento.	
<b>Codice:</b>	<b>Semestre:</b> I
<b>Prerequisiti / Propedeuticità:</b> Nessuno	
<b>Metodo didattico:</b>	
<b>Materiale didattico:</b>	
<b>Modalità di esame:</b> Scritto e orale	

<b>Insegnamento: Sistemi di Information Retrieval</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<p><b>Obiettivi formativi:</b> L'obiettivo del corso è quello di fornire un'introduzione ai concetti fondamentali e alle tecniche dell'Information Retrieval. Il corso presenterà metodi, modelli e tecnologie per la ricerca di informazioni e descriverà sia aspetti teorici che architetturali dei sistemi per l'information retrieval. Verranno presentate inoltre alcune applicazioni di queste tecnologie come l'estrazione automatica di informazioni e la loro organizzazione, i motori di ricerca su Web, i sistemi di categorizzazione e clusterizzazione.</p>	
<p><b>Contenuti:</b> L'accesso alle informazioni, data retrieval e information retrieval, sistemi per l'accesso a informazioni, classificazione di rilevanza. Interfacce di ricerca, navigazione e ricerca, specifica delle query, visualizzazione dei risultati. Modelli per l'information retrieval, booleano, vettoriale, probabilistico, modelli avanzati per l'information retrieval. Valutazione del processo di retrieval, precision/recall, f-measure, altre misure per la valutazione, collezioni di documenti. L'utente nel processo di retrieval, relevance feedback, query expansion. Retrieval di documenti testuali, proprietà, preprocessing, organizzazione dei documenti, richiami alla classificazione e al clustering. Il processo di indicizzazione. Web Retrieval, architetture dei motori di ricerca sul web, architettura base e clustered, caching, architetture distribuite, ranking, metadati, interazione con l'utente, browsing. Web Crawling, architetture dei sistemi di crawling, applicazioni, algoritmi di scheduling., cenni ai sistemi di raccomandazione e al multimedia information retrieval.</p>	
<b>Codice:</b>	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b> Nessuna	
<b>Metodo didattico:</b> Lezioni frontali, esercitazioni, seminari applicativi	
<b>Materiale didattico:</b> slides del corso, Libri di testo: Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Modern Information Retrieval: The Concepts and Technology Behind Search 2ed, Addison Wesley, 2011	
<b>Modalità di esame:</b> Sviluppo di un elaborato, prova orale	



<b>Insegnamento: Sistemi Real Time</b>	
<b>CFU:</b> 6	<b>SSD:</b> ING-INF/05
<b>Ore di lezione:</b> 38	<b>Ore di esercitazione:</b> 10
<b>Anno di corso:</b> II	
<b>Obiettivi formativi:</b>	
<p>Il corso fornisce le conoscenze di base sui sistemi in tempo reale, sulla schedulazione di task real-time, sulla gestione delle risorse, sulle reti di calcolatori e sui sistemi operativi adottati in ambito industriale. Fornisce inoltre le competenze necessarie alla progettazione, il dimensionamento e lo sviluppo di sistemi in tempo reale. Le esercitazioni consistono in applicazioni di programmazione concorrente con task real-time sviluppate in ambiente LINUX real-time (patch RTAI) e progettazione OO di software real-time attraverso il profilo OMG MARTE.</p>	
<b>Contenuti:</b>	
<p>Concetti Introduttivi. Introduzione ai sistemi in tempo reale: campi applicativi, l'esempio del controllo digitale, dimensionamento, deadline, sistemi hard e soft real-time, tempo reale e velocità; caratteristiche desiderabili: timeliness, prevedibilità, tolleranza ai sovraccarichi, monitorabilità, flessibilità; problematiche di progetto e sviluppo; ruolo dei sistemi operativi real time (RTOS). Prevedibilità di un sistema di calcolo. Fonti di non determinismo del hardware e del Sistema Operativo. Valutazione del WCET.</p> <p>Introduzione allo Scheduling. Processo e programma. Schedulazione, fattibilità, schedulabilità, ottimalità, preemption. Classificazione degli algoritmi di scheduling. Metriche di scheduling. Vincoli sui processi. Cenni allo scheduling di sistemi non real time.</p> <p>Scheduling di task real time. scheduling di task aperiodici: algoritmo di Jackson, algoritmo di Horn, algoritmo di Bratley. Scheduling con vincoli di precedenza. Scheduling di task periodici: timeline Scheduling, Rate Monotonic (RM). Earliest Deadline First (EDF), Deadline Monotonic. Ottimalità e test di garanzia. Response Time Analysis. Cenni al Processor Demand Criterion per EDF.</p> <p>Server aperiodici. Schedulazione in background. Polling Server (PS). Server aperiodici a priorità fissa: Deferrable Server (DS), Sporadic Server (SS), Slack Stealer. Server aperiodici a priorità dinamica: Dynamic Sporadic Server (DSS), Total Bandwidth Server (TBS), Costant Bandwidth Server (CBS).</p> <p>Accesso a risorse condivise. Il problema della priority inversion. Non-preemptive protocol. Highest locker priority. Priority Inheritance e Priority Ceiling: bloccaggio diretto e indiretto, analisi di schedulabilità e test di garanzia, calcolo dei tempi di bloccaggio.</p> <p>Gestione dei sovraccarichi. Carico, valore cumulativo. Gestione di sovraccarichi transitori: admission control; robust scheduling, resource reservation con CBS. Algoritmo RED: schedulabilità, strategia di rigetto e resource reclaiming. Gestione di sovraccarichi permanenti: job skipping, period adaptation e service adaptation.</p> <p>Comunicazione Real-time. I protocolli CSMA/CD e Token Ring. Modelli di traffico real-time. Fonti di non determinismo nelle reti. Soluzioni per la comunicazione real-time a livello data link: Controller Area Network (CAN), ProfiNET ed Ethernet Power Link.</p> <p>Real Time Operating Systems (RTOS): Standard ed esempi.</p> <p>Real Time Application Interface (RTAI): primitive ed esercitazioni.</p> <p>Progettazione OO di Sistemi Real Time con OMG MARTE. Esempi di progettazione in ambiente integrato (Papyrus).</p>	
<b>Codice:</b> 31682	<b>Semestre:</b> II
<b>Prerequisiti / Propedeuticità:</b>	
<b>Metodo didattico:</b> lezioni e esercitazioni in aula, attività di laboratorio	
<b>Materiale didattico:</b> trasparenze dalle lezioni del corso, libro di testo: G. Buttazzo "Sistemi in tempo reale", materiale esercitativo, risorse su rete.	
<b>Modalità di esame:</b> prova scritta (opzionalmente basata su esercizi di progettazione/programmazione assegnati durante il corso) e colloquio orale.	

<b>Insegnamento: Specifica di sistemi</b>	
<b>CFU: 6</b>	<b>SSD: INF/01</b>
<b>Ore di lezione: 38</b>	<b>Ore di esercitazione: 10</b>
<b>Anno di corso: II</b>	
<b>Obiettivi formativi:</b> Il corso si propone di fornire le nozioni di base per il problema della modellizzazione formale di sistemi Hardware e Software finalizzate alla verifica delle proprietà di correttezza. In particolare, il corso riguarderà la modellizzazione di sistemi a stati finiti (automi) o infiniti (sistemi pushdown e real-time) sia "chiusi" (non interagenti con l'ambiente) che "aperti"(interagenti con l'ambiente). Per i sistemi aperti, in particolare, verrà considerata come tecnica di modellazione la teoria dei giochi e il module checking.	
<b>Contenuti:</b> Automi a stati finiti su parole finite e infinite e loro problemi decisionali. Automi a stati finiti su alberi finiti e infiniti e loro problemi decisionali. Automi gerarchici. Reti di Petri. Automi pushdown su parole e alberi infiniti e loro problemi decisionali. Formalismi per sistemi real time (Timed automata). Nozioni di teoria dei giochi per la verifica di sistemi interagenti con l'ambiente: giochi su sistemi a stati finiti; giochi con informazione parziale; module checking su sistemi a stati finiti e infiniti.	
<b>Codice:</b>	<b>Semestre: I</b>
<b>Prerequisiti / Propedeuticità:</b> Nessuno	
<b>Metodo didattico:</b>	
<b>Materiale didattico:</b>	
<b>Modalità di esame:</b> Orale	

## **Calendario delle attività didattiche - a.a. 2016/2017**

	<b>Inizio</b>	<b>Termine</b>
<b>1° periodo didattico</b>	20 settembre 2016	16 dicembre 2016
<b>1° periodo di esami</b> <sup>(a)</sup>	17 dicembre 2016	4 marzo 2017
<b>2° periodo didattico</b>	6 marzo 2017	9 giugno 2017
<b>2° periodo di esami</b> <sup>(a)</sup>	10 giugno 2017	31 luglio 2017
<b>3° periodo di esami</b> <sup>(a)</sup>	29 agosto 2017	30 settembre 2017

(a): per allievi in corso

### **Referenti del Corso di Studi**

Coordinatore Didattico dei Corsi di Studio in Ingegneria Informatica: Prof. Carlo Sansone – Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - tel. 081/7683640 - e-mail: carlo.sansone@unina.it.

Referente del Corso di Laurea per il Programma ERASMUS+: Prof. Simon Pietro Romano – Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - tel. 081/7683823 - e-mail: spromano@unina.it.

Referente del Corso di Laurea per i tirocini: Prof. Antonio Pescapè – Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - tel. 081/7683856 – e-mail: pescape@unina.it.

### **Sito web del Corso di Studi**

<http://www.ingegneria-informatica.unina.it>

### **Requisiti curriculari minimi per l'accesso alla Laurea Magistrale in Ingegneria Informatica (LM-32)**

Lo studente in possesso del titolo di Laurea ex D.M. 509/99 o ex D.M. 270/04 potrà essere ammesso al Corso di Laurea Magistrale in Ingegneria Informatica se avrà acquisito nella precedente carriera CFU nei settori scientifico disciplinari di seguito indicati nella misura minima corrispondentemente indicata:

<b>SSD</b>	<b>CFU minimi</b>
MAT/05, MAT/03, MAT/09, FIS/01, FIS/03, MAT/08	24
ING-INF/01, ING-INF/02, ING-INF/03, ING-INF/04, ING-INF/05, ING-INF/06, ING-INF/07, ING-IND/13, ING-IND/16, ING-IND/17, ING-IND/31, ING-IND/32, ING-IND/34, ING-IND/35, INF/01	54