



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

**SCUOLA DI SCIENZE**

**Bollettino Notiziario**

Anno Accademico 2018/2019

**Laurea magistrale in Informatica (Ord. 2014)**

---

# Curriculum: Corsi comuni

---

## ADVANCED TOPICS IN COMPUTER SCIENCE

(Titolare: Dott. MASSIMILIANO DE LEONI)

**Periodo:** I anno, annuale  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A; 6,00 CFU

### Prerequisiti :

Nessun prerequisito

### Conoscenze e abilità da acquisire :

Questo corso verte sulla modellazione, l'analisi di processi aziendali e sull'installazione e la configurazione di sistemi informativi di supporto a questi processi. Alla fine del corso, gli studenti sono in grado di:

• modellare processi aziendali reali;

• interpretare i requisiti formali e tradurli in modelli espliciti;

• analizzare e migliorare i processi (e i relativi sistemi che li supportano) attraverso simulazione e analisi di sensibilità;

• installare e configurare un sistema informativo conforme ai vincoli imposti dal modello di processo.

### Attività di apprendimento previste e metodologie di insegnamento :

L'obiettivo ultimo di ogni sistema informativo è quello di supportare processi aziendali. Per questa ragione, i sistemi informativi devono essere disegnati ed analizzati in modo tale da non violare certe normative (per esempio, questi devono seguire alcuni requisiti legali e di auditing). Anche i tempi di risposta e di esecuzione devono essere i più brevi possibili, così come i costi devono essere ridotti e i rischi minimizzati.

Sistemi aziendali come i sistemi di Workflow Management, Enterprise Resource Planning (ERP), Business Process Management (BPM), Enterprise Information (EI), Customer Relationship Management (CRM) e di Product Data Management (PDM) sono sistemi generici che sono configurati sulla base di modelli di processo.

Alcuni sistemi implementano processi monolitici isolati, mentre altri sistemi permettono la composizione di diversi servizi (anche web services) fino ad ottenere processi della giusta complessità.

In alcuni sistemi, i modelli di processo sono espliciti e possono essere adattati (per esempio, in sistemi di BPM), mentre, in altri, sono impliciti (per esempio, i reference models nei sistemi SAP).

In tutti i casi, è chiaro che processi e sistemi sono strettamente intrecciati in qualsiasi azienda. Per questa ragione, è importante che gli studenti capiscano la relazione tra sistemi e processi e siano in grado di modellare sistemi complessi che coinvolgono, processi, servizi, umani ed diverse organizzazioni.

Il corso verte sull'intero ciclo di vita di progettazione, configurazione ed installazione di sistemi informativi aziendali guidati da processi. In particolare, il corso tocca i seguenti aspetti:

1. interpretazione di requisiti di processo informali e la loro traduzione in espliciti modelli formali;

2. analisi delle performance dei processi attraverso simulazione ed analisi di sensibilità;

3. il miglioramento dei processi (e dei modelli corrispondenti) attraverso identificazione e la risoluzione di, per esempio, colli di bottiglia, sovra/sottoutilizzazione di risorse, riduzione di costi e tempi di servizio

4. la configurazione di sistemi informativi che vengono utilizzati in conformità con tali modelli.

Le reti di Petri sono utilizzate nel corso per modellare processi, in due varianti: colorate e YAWL. A partire da descrizioni informali dei processi, durante il corso, gli studenti modelleranno i processi in CPN Tools e faranno simulazioni ed analisi per la loro ottimizzazione. I modelli di processo così migliorati verranno alla fine installati in YAWL, un sistema informativo aziendale open-source.

### Contenuti :

Il corso si compone di cicli di lezioni monografici, che illustrano temi avanzati dell'informatica con il supporto di esperti internazionali.

### Modalità di esame :

Tesina. Gli studenti ricevono una descrizione testuale di un processo realistico che devono modellare ed ottimizzare. Successivamente, questi devono configurare un sistema informativo in modo tale che il suo funzionamento sia conforme al processo modellato.

La tesina è divisa in tre parti da sottomettere nell'ordine giusto ed entro le scadenze stabilite.

### Criteri di valutazione :

La tesina verifica le capacità degli studenti di modellare, ottimizzare processi aziendali e di configurare sistemi informativi conformi.

### Testi di riferimento :

Aalst, Wil M. P. : van der; Stahl, Christian, Modeling business processes a Petri net-oriented approach Wil van der Aalst and Christian Stahl. Cambridge: MIT Press, 0

Arthur H.M. ter Hofstede, Wil M.P. van der Aalst, M. Adams, N. Russell., Modern Business Process Automation: YAWL and its Support Environment. : Springer-Verlag Berlin Heidelberg, 2010

### Eventuali indicazioni sui materiali di studio :

In aggiunta a testi di riferimento (vedi sotto), materiale aggiuntivo è disponibile su <http://cpntools.org/> sull'uso di CPN Tools e del suo linguaggio. Su <http://www.yawlfoundation.org/>, è possibile trovare i manuali di uso e di configurazione di YAWL.

---

## ALGORITMI AVANZATI

(Titolare: Dott. DAVIDE BRESOLIN)

**Periodo:** I anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

### Prerequisiti :

Il corso richiede familiarità con alcuni concetti algoritmici di base, quali complessità asintotica, algoritmi di ricerca e ordinamento e

strutture dati di base come alberi, liste, mappe, vettori e matrici.

Non ci sono corsi propedeutici.

**Conoscenze e abilità da acquisire :**

Il Pensiero Algoritmico Ã quel processo che permette di analizzare e risolvere problemi computazionali anche molto complessi ad un livello di astrazione indipendente dal particolare linguaggio o sistema informatico che si sta utilizzando. Il corso si pone l'obiettivo di insegnare agli studenti a pensare "in modo Algoritmico", fornendo le seguenti conoscenze e abilitÃ attese:

1. Essere in grado di comprendere un problema da risolvere, di quali sono i dati disponibili e come deve essere fornita la soluzione.
2. Essere in grado di formulare un problema in termini di input e output trattabili da un calcolatore.
3. Essere in grado di definire un algoritmo e di analizzarne le proprietÃ di correttezza e complessitÃ computazionale.
4. Essere in grado di implementare un algoritmo in un linguaggio di programmazione concreto.
5. Essere in grado eseguire l'implementazione su un dataset di media dimensione e di comprenderne ed analizzarne i risultati.

**AttivitÃ di apprendimento previste e metodologie di insegnamento :**

Il corso Ã suddiviso in blocchi tematici, ognuno dei quali affronta un problema reale e lo risolve seguendo l'approccio del pensiero algoritmico. Ogni blocco inizia con una serie di lezioni frontali in aula durante le quali vengono affrontati i contenuti del corso. Terminata l'attivitÃ frontale, il blocco prosegue con una o piÃ lezioni di laboratorio dove gli studenti divisi a gruppi implementeranno e testeranno la soluzione del problema reale su un dataset di media dimensione, e si conclude con una fase di confronto e discussione delle varie soluzioni realizzate.

**Contenuti :**

Introduzione al pensiero algoritmico. Grafi: nozioni di base e rappresentazione. Generazione randomizzata di grafi. Visite in ampiezza e in profonditÃ di un grafo. Componenti Connesse. Algoritmi "greedy". Grafi pesati: cammini minimi e alberi di connessione minimi. Strutture dati per insiemi disgiunti. Problemi trattabili e problemi intrattabili. Problemi NP-completi e algoritmi di approssimazione. Il problema del Commesso Viaggiatore: soluzioni esatte ed euristiche di approssimazione. Algoritmi "divide et impera": algoritmi di clustering. Programmazione dinamica: similaritÃ e allineamento di sequenze.

**ModalitÃ di esame :**

L'esame Ã diviso in una parte di teoria ed una parte pratica.

La parte di teoria consiste in una prova scritta che si tiene nella normale sessione d'esame.

La parte pratica puÃ essere svolta in due modalitÃ diverse:

- DURANTE IL CORSO: un gruppo di massimo tre studenti implementa tutti gli algoritmi visti nei laboratori e consegna il codice ed i risultati ottenuti.

- DOPO IL CORSO: un unico progetto personale, da sviluppare seguendo i cinque passi del Pensiero Algoritmico. In questo caso il progetto va svolto da soli, non in gruppo.

Il voto finale Ã la media aritmetica dei voti delle due parti.

**Criteri di valutazione :**

I criteri di valutazione con cui verrÃ effettuata la verifica delle conoscenze e delle abilitÃ acquisite sono:

1. Completezza delle conoscenze acquisite;
2. ProprietÃ della terminologia tecnica utilizzata;
3. CapacitÃ di formalizzare un problema in termini algoritmici
3. CapacitÃ di analizzare correttezza e complessitÃ computazionale di un algoritmo
4. CapacitÃ di implementare un algoritmo in modo coerente con la teoria
5. CapacitÃ di utilizzare un algoritmo per risolvere un problema concreto

**Testi di riferimento :**

Cormen, Thomas H.; Colussi, Livio, *Introduzione agli algoritmi e strutture dati* Thomas Cormen ... [et al.] edizione italiana a cura di Livio Colussi. Milano [etc.]: McGraw-Hill, 2010

Cormen, Thomas H., *Introduction to algorithms* Thomas H. Cormen ... [et al.]. Cambridge (Mass.): MIT Press, 2009

**Eventuali indicazioni sui materiali di studio :**

Il corso ha una sezione dedicata sul Moodle del Dipartimento di Matematica. Il Moodle raccoglie le dispense del corso, le specifiche dettagliate dei singoli laboratori, gli esercizi e le loro soluzioni. VerrÃ usato anche per comunicazioni e aggiornamenti da parte del Docente.

---

## ALTRE CONOSCENZE UTILI PER IL LAVORO

(Titolare: Dott. DAVIDE BRESOLIN)

**Periodo:** Il anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** ; 3,00 CFU

---

## AMMINISTRAZIONE DI SISTEMA

(Titolare: Dott. FRANCESCO CLABOT)

**Periodo:** I anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A+8E; 6,00 CFU

**Prerequisiti :**

Il corso non prevede particolari prerequisiti.

**Conoscenze e abilitÃ da acquisire :**

Il corso si propone di presentare agli studenti lâ€™organizzazione di un dipartimento ICT di una grande azienda. In particolare verranno trattate tematiche legate alle metodologie consolidate per lâ€™impostazione dei processi ICT (ITIL), le motivazioni che sono alla base delle scelte dei prodotti e tecnologie adottate (ROI, SLA, etc.), esempi concreti di architetture informatiche basilari oltre a vari case studies.

**AttivitÃ di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali e laboratorio

### **Contenuti :**

- La gestione dei servizi informatici (ITIL): i processi coinvolti nelle due aree della Gestione dei Servizi (Service Support e Service Delivery), la loro applicazione al ciclo operativo completo dei servizi, gli obiettivi fondamentali e perché questi sono stati standardizzati, breve dissertazione su ognuno dei 10 servizi coinvolti, esempi pratici.

- Modelli di servizio: considerazioni su ROI e SLA, approccio ed aspetti pratici.

- Il dipartimento IT: struttura ed organizzazione. Organigramma generale e breve dissertazione sui vari settori. Analisi accurata del "Service Desk" (come evoluzione dell'Help Desk).

- L'infrastruttura informatica: in verticale dal network ai servizi richiamando sempre i concetti esposti nella prima parte del corso. Esempi pratici (no laboratorio) e case study per mettere alla prova le capacità deduttive degli studenti

### **Modalità di esame :**

Test scritto e prova di laboratorio.

### **Criteri di valutazione :**

Le conoscenze dello studente vengono valutate mediante un test a risposta multipla. La votazione finale prenderà in considerazione anche la qualità dell'attività di laboratorio condotta.

### **Testi di riferimento :**

CONTENUTO NON PRESENTE

### **Eventuali indicazioni sui materiali di studio :**

Sul sito web del corso (link da <http://www.netadm.it>) sono presenti molti documenti scaricabili in formato digitale: case study, articoli divulgativi etc.

## **APPRENDIMENTO AUTOMATICO**

(Titolare: Prof. FABIO AIOLLI)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A+8L; 6,00 CFU

### **Prerequisiti :**

È opportuno avere familiarità con le conoscenze matematiche relative al Calcolo delle Probabilità e all'Analisi di funzioni multivariate. Inoltre è consigliabile avere conoscenze di base relative alla Programmazione e all'Intelligenza Artificiale.

L'insegnamento non prevede propedeuticità.

### **Conoscenze e abilità da acquisire :**

In questo insegnamento si presentano alcuni dei concetti fondamentali che caratterizzano l'Apprendimento Automatico, cioè quella classe di tecniche ed algoritmi che a partire da dati empirici permettono di acquisire nuova conoscenza, oppure di correggere e/o raffinare conoscenza già disponibile. Tali tecniche sono particolarmente utili per problemi per cui è impossibile o molto difficile pervenire ad una formalizzazione utilizzabile per la definizione di una soluzione algoritmica ad hoc. Esempi di tali problemi sono compiti percettivi, come il riconoscimento visivo di cifre manoscritte, e problemi in cui i dati sono corrotti dal rumore o sono incompleti. L'insegnamento tratta principalmente metodi numerici.

Sono previste esercitazioni in laboratorio informatico che consentono allo studente di sperimentare le conoscenze acquisite mediante l'applicazione a piccoli esempi pratici.

### **Attività di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali ed esercitazioni in laboratorio informatico. Le esercitazioni in laboratorio informatico consistono nella sperimentazione da parte degli studenti delle tecniche viste a lezione sotto vari scenari operativi. In questo modo gli studenti possono verificare sperimentalmente i concetti appresi e acquisire sia capacità di applicazione dei concetti appresi che di giudizio critico.

### **Contenuti :**

La struttura e le tematiche dell'insegnamento saranno le seguenti:

- Introduzione:

Quando Applicare le Tecniche Proprie dell'Apprendimento Automatico; Paradigmi di Apprendimento Automatico; Gli ingredienti Fondamentali dell'Apprendimento Automatico.

- Apprendimento di Concetti:

Complessità dello Spazio delle Ipotesi; Misure di Complessità; Esempi di Algoritmi di Apprendimento Supervisionato;

- Alberi di Decisione:

Apprendimento di Alberi di Decisione; Trattamento di Dati Numerici, di Dati Mancanti, di Costi; Tecniche di Pruning e Derivazione di Regole di Decisione.

- Apprendimento Probabilistico:

Apprendimento Bayesiano; Esempi di Applicazione al Paradigma Supervisionato e al Paradigma Non-Supervisionato (clustering); Classificatore Ottimo di Bayes; EM.

- Reti Neurali e Support Vector Machines:

Cenni di Reti Neurali; Margine di Classificazione; Support Vector Machines per Classificazione e Regressione; Funzioni Kernel.

- Aspetti Applicativi:

Pipeline di Classificazione; Rappresentazione e Selezione di Variabili Categoriche; Model Selection, Holdout, CrossValidation, LeaveOneOut CV; Criteri Esterni e Interni per Valutare un Sistema di Clustering; Sistemi di Raccomandazione: Tipologie, Approcci, Misure di Valutazione.

### **Modalità di esame :**

Lo studente deve superare un esame scritto e, se ritenuto necessario dal docente, un esame orale.

### **Criteri di valutazione :**

Il testo dell'esame scritto contiene alcune domande che consentono di valutare il livello di apprendimento delle nozioni impartite durante l'insegnamento e la capacità dello studente nell'analizzarle criticamente. Sono poi presenti domande in cui si richiede allo studente di mostrare di aver compreso gli aspetti applicativi trattati all'interno delle attività svolte in laboratorio informatico. Tali domande hanno lo scopo di valutare se lo studente ha sviluppato la capacità di applicare le nozioni apprese durante l'insegnamento.

Nel caso in cui la valutazione dello scritto non risulti soddisfacente per lo studente, il docente può integrare l'esame scritto con un

esame orale per meglio verificare la preparazione dello studente.

**Testi di riferimento :**

Alpaydin, Ethem, *Introduction to machine learning*. Cambridge: The MIT press, 2010

Mitchell, Tom M., *Machine learning*. New York: McGraw-Hill, 1998

**Eventuali indicazioni sui materiali di studio :**

Vengono rese disponibili, come riferimento, i lucidi utilizzati a lezione.

---

## ASPETTI AVANZATI DEI LINGUAGGI DI PROGRAMMAZIONE

(Titolare: Dott.ssa SILVIA CRAFA)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Conoscenze di programmazione e di programmazione ad oggetti.

**Conoscenze e abilità da acquisire :**

Il corso presenta alcune tecniche avanzate dei moderni linguaggi di programmazione

ed affronta il tema più generale dell'analisi dell'impatto sociale delle nuove tecnologie informatiche.

Lo studente svilupperà la capacità di comprendere, ragionare e valutare alcune delle nuove tecniche di programmazione e svilupperà la capacità di analisi critica del ruolo della tecnologia nella società contemporanea.

**Attività di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali con esercizi ed approfondimenti di argomenti di ricerca tramite articoli scientifici.

**Contenuti :**

Il corso presenta alcune tecniche avanzate dei moderni linguaggi di programmazione, tra cui: l'uso dei sistemi di tipi per ragionare sui programmi, concetti avanzati di programmazione orientata agli oggetti (typing strutturale, type checking dinamico, mixins),

l'integrazione della programmazione funzionale con la programmazione ad oggetti, cenni di programmazione concorrente asincrona e reattiva. Questi concetti saranno affrontati in modo fondazione e visti all'opera in uno studio ragionato dei linguaggi Scala e Java8.

Nella parte finale del corso gli studenti saranno esposti all'analisi e alla discussione di alcuni casi concreti di riflessione sull'impatto sociale delle nuove tecnologie.

**Modalità di esame :**

Sono previste una prova scritta e una seconda prova che consiste nella discussione orale di un tema di approfondimento o in alternativa nella realizzazione di un progetto software.

**Criteri di valutazione :**

La prova scritta valuta l'acquisizione dello studente degli aspetti fondazionali affrontati durante il corso. La seconda prova valuta la capacità dello studente di analizzare e valutare aspetti avanzati delle tecnologie informatiche e del loro impatto sulla società.

**Testi di riferimento :**

B.C. Pierce, *Types and Programming Languages*. : The MIT Press, 2002

M. Odersky, L. Spoon, B. Venners, *Programming in Scala*. : Artima, 2016

---

## BIG DATA COMPUTING

(Titolare: Prof. ANDREA ALBERTO PIETRACAPRINA) - Mutuato da:

**Periodo:** I anno, 2 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Il corso ha i seguenti prerequisiti: competenze relative al progetto e all'analisi di algoritmi e strutture dati, conoscenza delle nozioni fondamentali di calcolo delle probabilità e statistica, e capacità di programmazione in Java o Python.

**Conoscenze e abilità da acquisire :**

In questo corso gli studenti imparano tecniche algoritmiche fondamentali per l'elaborazione efficiente ed efficace di insiemi di dati di grande dimensione. Inoltre, attraverso alcune attività pratiche, essi acquisiscono abilità relative allo sviluppo di applicazioni in Apache Spark, che è uno dei framework di programmazione più popolari e diffusi per big data computing.

**Attività di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali e attività propedeutiche allo svolgimento degli homework.

**Contenuti :**

Il corso affronterà i seguenti argomenti:

Introduzione al fenomeno dei Big Data

Programming frameworks: MapReduce/Hadoop, Spark

Clustering

Association Analysis

Graph Analytics (metriche di centralità, scale-free/Power-law graphs, fenomeno dello small world, uncertain graphs)

Similarity and diversity search

**Modalità di esame :**

L'esame consiste in alcuni homework di programmazione, assegnati ogni 2-3 settimane e da svolgere in gruppi di 3-4 studenti, e in una prova scritta individuale comprendente domande teoriche ed esercizi.

**Criteri di valutazione :**

La valutazione finale è basata sugli homework e sulla prova scritta. Gli homework mirano a verificare la capacità degli studenti di programmare applicazioni big data in Apache Spark, mentre la prova scritta mira a verificare la loro conoscenza delle tecniche

algoritmiche apprese durante il corso e la loro capacità di problem solving nel contesto big data.

**Testi di riferimento :**

J. Leskovec, A. Rajaraman and J. Ullman, *Mining Massive Datasets.* : Cambridge University Press, 2014

**Eventuali indicazioni sui materiali di studio :**

Il diario delle lezioni, il materiale didattico e le modalità d'esame dettagliate sono resi disponibili sul sito web accessibile anche da MOODLE:

<http://www.dei.unipd.it/~capri/BDC/>

---

## BIOINFORMATICA

(Titolare: Prof. GIORGIO VALLE)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A+8E; 6,00 CFU

**Prerequisiti :**

Non ci sono prerequisiti particolari, se non quanto ci si aspetta da uno studente magistrale di informatica. Una conoscenza di base della genetica e della biologia molecolare saranno comunque utili per meglio inquadrare le motivazioni biologiche che stanno alla base della bioinformatica.

Il corso Ã in lingua inglese, quindi Ã necessario avere una buona conoscenza dell'inglese scritto e parlato.

**Conoscenze e abilita' da acquisire :**

Il Corso Ã suddiviso in tre parti principali: la prima parte mette in relazione Biologia e Informazione; la seconda parte descrive i principali algoritmi utilizzati in bioinformatica per allineare sequenze biologiche e assemblare genomi; la terza parte tratta di problemi di bioinformatica relativi alla genomica funzionale. Inoltre il corso Ã accompagnato da esercitazioni pratiche in cui gli studenti applicheranno metodi bioinformatici per analizzare dati genomici.

In considerazione della complessitÃ della materia e in accordo con i descrittori di Dublino, particolare attenzione sarÃ dedicata affinchÃ gli studenti acquisiscano la capacitÃ di integrare le conoscenze e gestire la complessitÃ dei problemi trattati, nonchÃ di formulare giudizi sulla base di informazioni limitate e spesso frammentarie.

**AttivitÃ di apprendimento previste e metodologie di insegnamento :**

Il corso sarÃ tenuto con lezioni frontali e con esercitazioni pratiche. L'attivitÃ didattica sarÃ supportata da risorse messe a disposizione sulla piattaforma e-learning "Moodle", che comprendono materiale per apprendimento remoto e per auto-valutazione. In questo modo si vuole promuovere un'attivitÃ di "blended learning" con cui lo studente, almeno in parte, impara autonomamente, seguendo un percorso che lo accompagna attraverso contenuti reperibili in rete. Dove possibile si applicherÃ il paradigma della "flipped classroom" che inverte lo schema tradizionale di insegnamento, prevedendo che prima lo studente impari la lezione autonomamente per poi discutere e approfondire gli argomenti in classe, con il docente e con gli altri studenti. Un'ampia raccolta di problemi, questionari ed esercizi viene messa a disposizione sulla piattaforma Moodle, sia per consentire l'autovalutazione, sia per stimolare argomenti di discussione da approfondire in classe.

**Contenuti :**

Questo Ã un corso di 6 crediti: cinque di lezioni ed uno di attivitÃ pratiche che consistono nell'implementazione di algoritmi oppure in un'approfondita indagine della letteratura, su argomenti assegnati.

Le lezioni sono organizzate in tre parti.

La prima parte Ã un'approfondita introduzione alla Biologia, presentata come una disciplina scientifica centrata sull'Informazione. I meccanismi che facilitano la trasmissione e l'evoluzione dell'informazione biologica saranno presi come spunto per introdurre alcuni problemi della biologia che richiedono approcci computazionali e strumenti bioinformatici.

La seconda parte del corso descrive i principali algoritmi utilizzati per allineare sequenze biologiche, inclusi quelli sviluppati per il sequenziamento di DNA di ultima generazione. Sono inoltre descritti gli algoritmi utilizzati per l'assemblaggio "de novo" di genomi. Infine, la terza parte del corso copre alcuni aspetti della bioinformatica relativi alla genomica funzionale, come l'analisi del trascrittoma, le predizioni e annotazione genica, la ricerca di pattern e motivi per la predizione delle strutture proteiche. Inoltre viene discusso il ruolo della bioinformatica nell'analisi di genomi individuali e nella medicina personalizzata.

**ModalitÃ di esame :**

L'esame si articola in tre parti: 1) una sessione pratica nella quale lo studente deve descrivere un progetto di analisi di dati che deve essere consegnato almeno due giorni prima della data dell'esame, 2) una sessione di quiz su Moodle, che si svolgerÃ all'inizio dell'appello d'esame e 3) una discussione orale in cui lo studente deve descrivere il progetto che ha realizzato e rispondere a domande sui contenuti del corso. Un continuo monitoraggio sarÃ attuato durante l'intera durata del corso per verificare la comprensione degli studenti.

**Criteri di valutazione :**

Nell'esame finale gli studenti dovranno dimostrare una comprensione sistematica del settore e dovranno sapersi destreggiare con i metodi della ricerca associati ad esso. Inoltre gli studenti dovranno essere capaci di analisi critica, di valutare e sintetizzare idee nuove e complesse, integrando gli argomenti di questo corso con altre conoscenze.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Non sono previsti libri ufficiali di testo e gli studenti saranno stimolati a trovare le informazioni su fonti multiple. Il materiale didattico sarÃ messo a disposizione sulla piattaforma e-learning Moodle.

---

## COGNITIVE SERVICES

(Titolare: Dott. LAMBERTO BALLAN)

**Periodo:** I anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 32A+16L; 6,00 CFU

**Prerequisiti :**

Lo studente deve avere conoscenze di base di programmazione e algoritmi.  $\hat{A}$  inoltre consigliabile conoscere i concetti di base in termini di probabilità e di analisi delle funzioni multivariate.

**Conoscenze e abilità da acquisire :**

Questo corso insegna i concetti, i metodi e le tecnologie alla base dei Servizi Cognitivi, vale a dire API, SDK e servizi, tipicamente disponibili nella nuvola (cloud), che aiutano gli sviluppatori software a creare applicazioni di intelligenza artificiale. Esempi di funzioni intelligenti che possono essere aggiunte ad un'applicazione tramite l'utilizzo di Servizi Cognitivi sono: il rilevamento delle emozioni da video; riconoscimento facciale, del contenuto visivo e vocale; comprensione linguistica e del parlato.

Il corso inoltre insegna le competenze e le abilità specifiche necessarie per applicare tali concetti alla progettazione e all'implementazione di applicazioni di intelligenza artificiale.

Gli studenti dovranno affrontare esercizi pratici in laboratorio informatico, in modo da provare l'applicazione delle conoscenze acquisite a piccoli esempi pratici.

**Attività di apprendimento previste e metodologie di insegnamento :**

Il corso consiste in lezioni e esercizi in laboratorio informatico. Gli esercizi in laboratorio informatico consentono agli studenti di sperimentare, in diversi scenari operativi, le tecniche introdotte a lezione. In questo modo gli studenti possono verificare sperimentalmente i concetti appresi in classe, acquisire la capacità di applicare i concetti appresi e di esprimere un giudizio critico.

**Contenuti :**

Il corso comprende gli argomenti elencati di seguito:

- Introduzione:

Dalla conoscenza umana ai servizi cognitivi intelligenti; Breve introduzione ai paradigmi di Intelligenza Artificiale e Apprendimento Automatico.

- Servizi cognitivi:

Concetti basilari; Lingua, Discorso e servizi di visione; Servizi e API principali (IBM Watson, Microsoft, Google Cloud); Tecnologie abilitanti.

- Problemi di apprendimento automatico e di applicazione:

Classificazione; Apprendimento delle rappresentazioni e selezione delle variabili categoriali; Processo di apprendimento e di valutazione; Misure di valutazione.

- Riconoscimento visivo:

"Insegnare ai computer a vedere": estrarre informazioni ricche da dati visivi; Sfide: perché la visione artificiale è difficile?; Progettare funzionalità visive efficaci; Apprendimento delle rappresentazioni nella visione artificiale; Comprensione delle immagini.

- Esercizi pratici:

Cosa c'è nella scatola? Come costruire una pipeline di riconoscimento visivo; Utilizzo di servizi cognitivi per il riconoscimento / comprensione delle immagini; Combinazione di diversi servizi in uno scenario multi-modale.

**Modalità di esame :**

Lo studente deve sviluppare, in accordo con il docente, un piccolo progetto applicativo. Inoltre, lo studente deve presentare una relazione scritta sul progetto svolto, in cui si discutono criticamente tutte le questioni trattate durante la sua realizzazione. Lo studente presenterà e discuterà il progetto e, se ritenuto necessario dal docente, affronterà un esame orale.

**Criteri di valutazione :**

Il lavoro di progetto e l'eventuale esame orale saranno valutati sulla base dei seguenti criteri: a) conoscenza da parte dello studente dei concetti, dei metodi e delle tecnologie alla base dei Servizi Cognitivi; b) capacità dello studente di padroneggiare la tecnologia di implementazione; c) capacità di sintesi, chiarezza e astrazione dello studente, come dimostrato dalla relazione scritta e dal progetto.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Le presentazioni mostrate durante le lezioni sono rese disponibili agli studenti come materiale di riferimento.

## COMPUTABILITÀ

(Titolare: Prof. PAOLO BALDAN)

**Periodo:** 1 anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Il corso richiede familiarità con alcuni concetti matematici di base, quali relazioni, funzioni, insiemi, cardinalità, ordini parziali, principi di induzione.

Non ci sono corsi propedeutici.

**Conoscenze e abilità da acquisire :**

Obiettivo del corso è quello di avvicinare lo studente ai temi classici della teoria della calcolabilità, completando e approfondendo alcune conoscenze algoritmiche acquisite nella laurea di primo livello. Partendo dall'esame matematico del concetto di procedimento effettivo, si studiano i limiti che tale nozione impone sulla classe delle funzioni effettivamente calcolabili da un algoritmo, con lo sviluppo di una teoria dell'indecidibilità e della ricorsione.

**Attività di apprendimento previste e metodologie di insegnamento :**

Il corso prevede lezioni frontali ed esercizi.

**Contenuti :**

Saranno sviluppati i seguenti temi:

- Algoritmi ed il concetto di procedimento effettivo. Macchine a registri (URM). Funzioni parziali ricorsive. Equivalenze tra modelli di calcolo. Universalità dei modelli di calcolo. Tesi di Church.

- Enumerazione delle funzioni calcolabili. Esistenza di funzioni non calcolabili: il metodo della diagonalizzazione. Il teorema del parametro. Programmi universali.

- Problemi decidibili, indecidibili e semidecidibili. Indecidibilità del problema della fermata. Metodo di riduzione. Esempi di altri problemi indecidibili.

- Insiemi ricorsivi e ricorsivamente enumerabili. Teoremi di Rice e di Rice-Shapiro.  
- Funzionali. Definizioni ricorsive. Ordinamenti parziali, funzioni monotone e punti fissi. Funzionali ricorsivi. Il teorema di Myhill-Sheperdson. Primo teorema di ricorsione. Secondo teorema di ricorsione.

**Modalita' di esame :**

L'esame consiste di una prova scritta, principalmente focalizzata sullo svolgimento di esercizi di teoria della computabilit .

**Criteri di valutazione :**

La prova scritta contiene esercizi atti a verificare la capacit  dello studente di utilizzare nozioni e tecniche dimostrative apprese durante il corso, per la soluzione di problemi nuovi.

**Testi di riferimento :**

Nigel Cutland, *Computability. An Introduction to Recursive Function Theory..* : Cambridge University Press, 1980

**Eventuali indicazioni sui materiali di studio :**

Pagina web:

<http://www.math.unipd.it/~baldan/Computabilita>

---

## COMPUTER AND NETWORK SECURITY

(Titolare: Prof. MAURO CONTI)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 40A; 6,00 CFU

**Prerequisiti :**

Il corso non prevede propedeuticit .

Tuttavia, sono consigliate conoscenze di base di reti, crittografia, e sistemi distribuiti (tipicamente acquisite nei corsi di Laurea in Informatica).

**Conoscenze e abilita' da acquisire :**

Acquisire concetti di base di sicurezza (e.g., Access Control, User Authentication, Malware, Attacchi DoS, Intrusion Detection/Prevention, Software and OS security, Trusted Computing) e conoscenze di sicurezza di sistema in ambiente Linux/Windows/Android, sicurezza di reti wireless/wired, web-application security.

Al termine del corso gli studenti saranno in grado di: progettare lâ€™architettura di sistemi ed applicazioni sicure, e aggiornare autonomamente le proprie competenze nel settore, anche tramite risultati recenti della ricerca nell'area.

**Attivit  di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali; discussione di articoli scientifici.

**Contenuti :**

- 1) COMPUTER SECURITY TECHNOLOGY AND PRINCIPLES: Cryptographic Tools, User Authentication, Access Control, Database Security, Malicious Software, Denial-of-Service Attacks, Intrusion Detection, Firewalls and Intrusion Prevention Systems.
- 2) SOFTWARE SECURITY AND TRUSTED SYSTEMS: Buffer Overflow, Software Security, Operating System Security, Trusted Computing and Multilevel Security.
- 3) MANAGEMENT ISSUES: IT Security Management and Risk Assessment, IT Security Controls, Plans, and Procedures, Physical and Infrastructure Security, Human Resources Security, Security Auditing, Legal and Ethical Aspects.
- 4) PART FOUR CRYPTOGRAPHIC ALGORITHMS: Symmetric Encryption and Message Confidentiality, Public-Key Cryptography and Message Authentication.
- 5) NETWORK SECURITY: Internet Security Protocols and Standards, Internet Authentication Applications, Wireless Network Security.

The second part of the course takes the form of seminars based on a selection of scientific papers (that either have had a strong impact on security today, or explore novel ideas that may be important in the future).

**Modalita' di esame :**

Scritta.

**Criteri di valutazione :**

Conoscenza dei concetti studiati nel corso.

**Testi di riferimento :**

W. Stallings, L. Brown, *Computer Security: Principles and Practice 2/E.* : Prentice Hall,

M. Bishop, *Introduction to Computer Security.* : Addison-Wesley Professional,

**Eventuali indicazioni sui materiali di studio :**

Libro (testo principale *Computer Security: Principles and Practice 2/E*) e articoli scientifici.

Il corso sar  tenuto in Inglese.

Il sito web del corso offrir  tutte le informazioni e materiale ulteriore:

<http://www.math.unipd.it/~conti/teaching.html>

---

## CRITTOGRAFIA

(Titolare: Prof. ALESSANDRO LANGUASCO) - Mutuato da: Laurea magistrale in Matematica (Ord. 2011)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 40A+8E; 6,00 CFU



**Prerequisiti :**

Gli argomenti dei corsi di Algebra (congruenze, gruppi e gruppi ciclici, campi finiti), Analisi I (calcolo differenziale ed integrale, serie numeriche) del corso di studi in Matematica.

**Conoscenze e abilità da acquisire :**

Lo scopo del corso è quello di offrire una panoramica delle basi teoriche necessarie per permettere uno studio critico dei protocolli crittografici usati oggi in molte applicazioni (autenticazione, commercio digitale). Nella prima parte verranno esposti gli strumenti matematici di base (essenzialmente dalla teoria elementare ed analitica dei numeri) necessari per comprendere il funzionamento dei moderni metodi a chiave pubblica. Nella seconda parte vedremo come applicare queste conoscenze per studiare in modo critico alcuni protocolli crittografici.

**Attività di apprendimento previste e metodologie di insegnamento :**

Lezione frontale.

**Contenuti :**

First Part: Basic theoretical facts: Modular arithmetic. Prime numbers. Little Fermat theorem. Chinese remainder theorem. Finite fields: order of an element and primitive roots. Pseudoprimality tests. Agrawal-Kayal-Saxena's test. RSA method: first description, attacks. Rabin's method and its connection with the integer factorization. Discrete logarithm methods. How to compute the discrete log in a finite field. Elementary factorization methods. Some remarks on Pomerance's quadratic sieve.

Second Part: Protocols and algorithms. Fundamental crypto algorithms. Symmetric methods (historical ones, DES, AES). Asymmetric methods. Attacks. Digital signature. Pseudorandom generators (remarks). Key exchange, Key exchange in three steps, secret splitting, secret sharing, secret broadcasting, timestamping. Signatures with RSA and discrete log.

**Modalità di esame :**

Esame scritto

**Criteri di valutazione :**

Durante la prova scritta lo studente dovrà rispondere ad alcune domande relative al programma svolto dimostrando di aver compreso gli argomenti del corso. Il massimo dei voti (30/30) verrà assegnato in presenza di un compito privo di errori. Il docente si riserva di fare alcune domande orali nel caso in cui sia necessario investigare ulteriormente la preparazione del candidato.

**Testi di riferimento :**

A. Languasco e A. Zaccagnini, *Manuale di Crittografia*. Milano: Hoepli, 2015

Koblitz, Neal, *A course in number theory and cryptography*. New York: Springer, 1987

**Eventuali indicazioni sui materiali di studio :**

Utilizzeremo i seguenti testi:

- 1) A.Languasco, A.Zaccagnini - *Manuale di Crittografia* - Hoepli Editore, 2015. (italian).
- 2) N.Koblitz - *A Course in Number Theory and Cryptography*, Springer, 1994.
- 3) R.Crandall, C.Pomerance - *Prime numbers: A computational perspective* - Springer, 2005.
- 4) B. Schneier - *Applied Cryptography* - Wiley, 1994

## DATA MINING

(Titolare: Dott.ssa MANUELA CATTELAN)

**Periodo:** I anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 34A+16L; 6,00 CFU

**Prerequisiti :**

Conoscenze di Informatica di base, Basi di Dati. Conoscenze di base di Probabilità e Statistica sono utili anche se non indispensabili.

**Conoscenze e abilità da acquisire :**

Ci si attende che lo studente acquisisca le seguenti conoscenze e abilità :

- padronanza dei principi di data mining per l'analisi di insiemi di dati anche ad elevata dimensionalità ;
- costruzione di modelli appropriati per l'analisi di insiemi di dati e la previsione;
- analisi dei dati tramite il software R, considerando sia analisi grafiche sia analisi di modellazione;
- valutazione ed interpretazione critica dei risultati ottenuti;
- capacità di comunicazione delle analisi dei dati effettuate e dei risultati conseguiti.

**Attività di apprendimento previste e metodologie di insegnamento :**

L'attività prevede

- 1) lezioni frontali, nelle quali vengono illustrati i contenuti del corso tramite l'utilizzo di slides relative agli aspetti teorici e all'analisi di insiemi di dati, con l'obiettivo di promuovere la discussione e la riflessione critica in aula;
- 2) esercitazioni in laboratorio informatico, per introdurre gli studenti all'analisi di insiemi di dati reali tramite l'utilizzo del software R.

**Contenuti :**

- Introduzione al corso: l'analisi dei dati come strumento di supporto per le decisioni. Motivazioni e contesto per il data mining.
- Modello lineare semplice e multiplo: stima, intervalli di confidenza, test di verifica di ipotesi, livello di significatività osservato, previsione, selezione del modello, analisi dei residui, correlazione spuria, multicollinearità
- Metodi di classificazione: regressione logistica, analisi discriminante lineare ed estensioni
- Criteri per la selezione del modello: validazione incrociata, R2 aggiustato, AIC, BIC, selezione automatica
- Regolarizzazione: regression ridge e lasso
- Regressione con componenti principali
- Regressione semiparametrica: splines di regressione, splines di liscio, modelli additivi generalizzati

**Modalità di esame :**

L'esame consiste in due prove scritte.

- 1) La prima prova (durata 1 ora) riguarda il modello di regressione lineare e consiste in una serie di domande a risposta multipla ed esercizi. Gli esercizi si riferiscono all'analisi di un insieme di dati reali e comprendono analisi numeriche, interpretazione di output da analisi effettuate in R e commenti di analisi grafiche. La prima prova si svolgerà dopo la metà del corso.

Durante la prima prova Ã" ammesso lâ€™ utilizzo di una calcolatrice tascabile.

2) La seconda prova si svolge in laboratorio informatico (durata 2 ore e 30 minuti) ed Ã" costituita dallâ€™ analisi di un insieme di dati reali tramite il software R. Si richiede la scrittura di un report che descriva lâ€™ analisi condotta, riportando le principali rappresentazioni grafiche e stime dei modelli e un'adeguata interpretazione dei risultati conseguiti.

Durante la prova lo studente puÃ² consultare il materiale del corso, quale il libro di testo, le slides, le dispense di laboratorio.

La valutazione finale della prova Ã" una media dei risultati delle due prove scritte.

Gli studenti che non effettueranno la prima prova a metÃ del corso potranno sostenerla immediatamente dopo la seconda prova in laboratorio.

#### **Criteri di valutazione :**

La prova dâ€™esame dâ€™esame ha lo scopo di

- 1) valutare le conoscenze acquisite relativamente alla costruzione e selezione di un modello di regressione lineare ed alla interpretazione critica dei risultati grafici e analitici conseguiti;
- 2) valutare le conoscenze acquisite relativamente all'applicazione di tecniche di modellazione appropriate per lâ€™ analisi di insieme di dati reali di natura diversa e per la previsione, con attenzione al caso di dati ad alta dimensionalitÃ ;
- 3) valutare le capacitÃ di utilizzo delle funzionalitÃ del software R per condurre una completa analisi di insieme di dati reali;
- 4) valutare le capacitÃ di interpretazione e comunicazione dei risultati dellâ€™ analisi di un insieme di dati reali.

#### **Testi di riferimento :**

Gareth, J., Witten, D., Hastie, T., Tibshirani, R., *An Introduction to Statistical Learning with Applications in R.* : Springer, 2013

#### **Eventuali indicazioni sui materiali di studio :**

Libro di testo. Sulla piattaforma Moodle sarÃ reso disponibile del materiale aggiuntivo, che include

- 1) le slides del corso;
- 2) le dispense di laboratorio informatico con R;
- 3) articoli e note dalla letteratura statistica e di data mining.

## **ECONOMIA DELL'INNOVAZIONE**

(Titolare: Prof.ssa FRANCESCA GAMBAROTTO)

**Periodo:** 1 anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

#### **Prerequisiti :**

Nessun prerequisito.

#### **Conoscenze e abilita' da acquisire :**

Il corso si pone due obiettivi: la comprensione del ruolo dellâ€™ innovazione nel processo economico capitalistico e la gestione aziendale delle attivitÃ innovative. Lâ€™ innovazione svolge un ruolo centrale nei processi di crescita economica e di sviluppo delle economie in quanto modifica la capacitÃ di utilizzo di risorse scarse e crea nuovi scenari organizzativi e distributivi di beni e servizi. Durante il corso vengono studiati e discussi i principi di funzionamento di unâ€™ economia basata sullâ€™ innovazione e i criteri manageriali di utilizzo dellâ€™ innovazione come input produttivo che come output di mercato.

Particolare attenzione sarÃ dedicata al ruolo delle tecnologie digitali diventate cosÃ pervasive nelle attivitÃ di produzione e di consumo allo scopo di discutere il loro ruolo nel cambiamento delle strategie aziendali nonchÃ degli effetti socio-economici che generano nelle economie avanzate.

#### **Attivita' di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali e attivitÃ di discussione in classe su argomenti specifici.

Utilizzo della piattaforma moodle come repository dei materiali utilizzati a lezione e per attivitÃ d'interazione.

#### **Contenuti :**

Gli argomenti trattati a lezione sono:

Economia dell'Innovazione: fonti dell'innovazione, innovazione e valore economico, strategie imprenditoriali, innovazione e dinamica dei mercati, effetti macroeconomici dell'innovazione tecnologica.

Management dell'innovazione: organizzazione dei processi d'innovazione, modelli di business, cooperazione ed ecosistemi per l'innovazione.

#### **Modalita' di esame :**

Prova scritta con domande chiuse ed aperte.

#### **Criteri di valutazione :**

Gli studenti frequentanti saranno valutati sugli argomenti svolti in aula e sulla partecipazione nelle attivitÃ di approfondimento tematico che verranno presentate all'inizio del corso.

Gli studenti non frequentanti saranno valutati sulla base della comprensione del manuale.

#### **Testi di riferimento :**

Schilling M. A., F. Izzo, *Gestione dell'innovazione.* Milano: McGraw Hill Education, 2017

#### **Eventuali indicazioni sui materiali di studio :**

Esame da Frequentanti:

il materiale del corso Ã" composto dai materiali di supporto utilizzati durante le lezioni, letture integrative e dai capitoli del manuale indicati a lezione.

Esame non frequentanti:

Il manuale di Schilling e Izzo riportato qui sotto.

## FUNCTIONAL LANGUAGES

---

(Titolare: Prof. GILBERTO FILE')

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A+10L; 6,00 CFU

### Prerequisiti :

Programmazione imperativa ed orientata agli oggetti

### Conoscenze e abilità da acquisire :

Apprezzare il paradigma funzionale di programmazione. In particolare viene studiato il linguaggio Haskell. Gli studenti imparano ad apprezzare l'importanza dei tipi come fondamentale mezzo per scoprire gli errori e imparano concetti quali il polimorfismo parametrico e la valutazione lazy. Viene introdotto anche il concetto di Monade e il suo uso per la programmazione generica e modulare. Imparano anche a capire la gestione run time dei programmi.

### Attività di apprendimento previste e metodologie di insegnamento :

Il corso prevede 50 ore totali. Di queste, circa 35 sono di lezioni frontali. Le restanti ore sono impiegate per la soluzione interattiva di esercizi e per la presentazione e discussione del progetto.

### Contenuti :

Nel corso viene illustrato il linguaggio funzionale Haskell. In particolare si studiano i seguenti aspetti:

Pattern matching.

Curryficazione e funzioni di ordine superiore.

Inferenza di tipo: cos'è e come viene fatta.

Input e output.

Polimorfismo parametrico.

Valutazione pigra.

Funtori, funtori applicativi e monadi.

Supporto run-time.

Parsing con le Monadi.

### Modalità di esame :

L'esame consiste di una parte scritta (80% del voto finale) e di una orale (20% del voto finale). L'esame scritto verte sulle nozioni generali e gli esercizi visti nel corso, mentre la parte orale è una discussione sul progetto che consiste nella realizzazione di un parser per un linguaggio funzionale.

### Criteri di valutazione :

L'esame mira a valutare il grado di comprensione dei concetti insegnati, raggiunto dagli studenti, e la capacità di usare questi concetti per affrontare problemi relativi ai linguaggi funzionali e al loro uso per definire programmi modulari e generici.

### Testi di riferimento :

Miran Lipovaca, Learning you a Haskell for great good. ;

Graham Hutton, Programming in Haskell. : Cambridge University Press,

### Eventuali indicazioni sui materiali di studio :

Programming in Haskell

by Graham Hutton

Cambridge University Press, 2nd edition

## GAME THEORY

---

(Titolare: Prof. LEONARDO BADIA) - Mutuato da:

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

### Prerequisiti :

Un corso anche basilare di teoria della probabilità.

### Conoscenze e abilità da acquisire :

L'insegnamento prevede l'acquisizione delle seguenti conoscenze e abilità, suddivise in due insiemi.

1: parte base. Apprendere e padroneggiare concetti teorici di base e avanzati della teoria dei giochi e saper risolvere problemi generali multi-obiettivo multi-agente con tecniche della teoria dei giochi.

2: parte applicativa. Sapere applicare i concetti della teoria dei giochi a scenari pratici, specialmente di tipo ICT; in questo contesto, e' di particolare interesse l'abilità di contestualizzare la teoria dei giochi come strumento di valutazione per l'efficacia della risoluzione tramite procedure multi-agente distributed.

A seconda del tipo di fruizione/frequenza/mutuazione del corso da parte dello studente, l'importanza di queste due insiemi viene considerata diversamente. In particolare, la parte applicativa è rilevante per studenti di corsi di ambito ingegneristico frequentanti. Per gli studenti di corsi differenti, o per studenti non frequentanti, l'acquisizione di conoscenze e abilità si concentra nell'insieme degli obiettivi di base.

### Attività di apprendimento previste e metodologie di insegnamento :

Lezioni convenzionali con il supporto di slide.

Prevista interazione su piattaforma moodle.

**Contenuti :**

Concetti base di teoria dei giochi  
Utilità, mercato, fattore di sconto  
Giochi statici in forma normale  
Dominanza, Equilibri di Nash  
Efficienza, prezzo dell'anarchia  
Giochi a somma zero, giochi minimax  
Strategie miste, equilibri misti  
Teorema di Nash, il teorema minimax  
The tragedy of the commons

Giochi dinamici  
Strategie e sottogiochi  
Backward utility  
Equilibri di Stackelberg  
Giochi ripetuti, collaborazione  
Duopoli dinamici, collusione  
Cooperazione, pricing  
Informazione incompleta/imperfetta  
Giochi bayesiani, signaling, beliefs  
Principio di rivelazione

Teoria dei giochi assiomatica  
Fictitious play  
Best response dynamics  
Ottimizzazione distribuita  
Game theory algoritmica  
Calcolo, complessità, e completezza dell'equilibrio  
Aste, bargaining  
Aste di primo e secondo prezzo  
Criterio VCG  
Giochi cooperativi, il nucleo, il valore di Shapley

Allocazione delle risorse  
Utilità, scelte e paradossi  
Giochi potenziali, coordinazione  
Algoritmi bio-inspired  
Giochi evolutivi  
Reti cognitive  
Selfish routing  
Sistemi multi-input con teoria dei giochi

**Modalità di esame :**

Per studenti di ingegneria frequentanti, l'esame coinvolge (a differenza degli altri) lo sviluppo di un progetto in gruppi di 1-3 persone, su argomenti del corso applicati alle ICT. L'argomento del progetto è concordato con il docente durante il corso.

Per tutti gli studenti in qualunque caso l'esame comprende anche un esame scritto obbligatorio a libro aperto, dove vengono sottoposti quattro diversi problemi di game theory allo studente su argomenti toccati durante il corso. Per ogni esercizio, vengono poste tre domande.

Gli studenti di ingegneria frequentanti si limiteranno a risolvere 3 esercizi sui 4 proposti. Tutti gli altri (studenti non frequentanti, o studenti non di ingegneria) dovranno risolverli tutti e 4.

Se il test scritto è sufficiente, gli studenti non frequentanti o gli studenti non di ingegneria potranno registrare il voto conseguito come voto finale dell'esame.

Gli studenti di ingegneria frequentanti invece discuteranno il progetto sviluppato durante il corso con un esame orale, da svolgersi dopo l'esame scritto. Questi esami orali si svolgono nella stessa giornata di un esame scritto, ma non necessariamente lo studente deve presentarsi nella stessa giornata per l'esame scritto e la discussione orale del progetto. Per il superamento dell'esame è necessaria una valutazione sufficiente di entrambe le parti, esame scritto e discussione orale.

**Criteri di valutazione :**

Ogni domanda nei test scritti viene valutata fino a un massimo di 3 punti.

Per gli studenti di ingegneria frequentanti, la discussione del progetto viene valutata fino a 10 punti.

Il voto finale è la somma numerica dei punteggi individuali delle domande e della discussione del progetto (se presente), limitata a 30.

Un punteggio di 30 e lode è assegnato agli studenti il cui punteggio numerico è superiore a 31.

Nella valutazione di ogni domanda scritta vengono tenuti in considerazione:

- la pertinenza, la correttezza, e la completezza della risposta;
- l'utilizzo appropriato delle terminologie, metodologie, e rappresentazioni formali tipiche della teoria dei giochi
- l'acquisita capacità di problem solving
- la capacità di discussione e verifica ex-post della soluzione trovata

Nella valutazione del progetto (se presente) vengono tenuti in considerazione:

- l'originalità della proposta e la pertinenza sia con le tematiche del corso che con le metodologie ingegneristiche tipiche dell'ICT
- la qualità dell'esposizione orale

- la capacita' di lavoro di gruppo e la presenza di singoli contributi attribuibili ai partecipanti al progetto
- la capacita' di trarre conclusioni significative dal punto di vista scientifico grazie alle metodologie apprese nel corso

**Testi di riferimento :**

S. Tadelis., *Game Theory: An Introduction..* : Princeton., 2013

A. MacKenzie, L. DaSilva, *Game Theory for Wireless Engineers.* : Morgan&Claypool, 2006

Noam Nisan, Tim Roughgarden, Eva Tardos, Vijay V. Vazirani (eds.), *Algorithmic Game Theory.* : Cambridge Univ. Press, 2007

Roberto Lucchetti, *A Primer in Game Theory.* : Esculapio, 2011

**Eventuali indicazioni sui materiali di studio :**

Diversi libri forniscono una trattazione generale di teoria dei giochi.

A mero titolo di suggerimento, si puo' usare il libro di Tadelis come riferimento in senso generale. Questa parte comunque dovrebbe essere integrata con materiale per le applicazioni. Il libro di MacKenzie e DaSilva e' un buon esempio, anche se non e' obbligatorio usare un libro per questo scopo (si puo' fare riferimento anche a materiale trovato in rete).

In ogni caso, il docente fornira' agli studenti tutte le dispense delle lezioni e appunti aggiuntivi.

---

## INTELLIGENZA ARTIFICIALE

(Titolare: Prof. ALESSANDRO SPERDUTI)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 40A+8L; 6,00 CFU

**Prerequisiti :**

E' opportuno possedere le conoscenze di base relative al Calcolo delle Probabilita', alla Programmazione e agli Algoritmi.

**Conoscenze e abilita' da acquisire :**

In questo insegnamento si presentano le tecniche fondamentali di alcuni degli approcci principali, all'interno della Intelligenza Artificiale, per la soluzione di problemi difficili. In particolare sono esaminate tecniche di Ricerca in uno Spazio di Soluzioni, Sistemi con vincoli, tecniche di Rappresentazione e Manipolazione di Conoscenza con e senza incertezza, di Pianificazione, e cenni di Apprendimento Automatico.

E' previsto lo sviluppo da parte del singolo studente, o di un gruppo di studenti, di un progetto.

**Attivita' di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali.

**Contenuti :**

La struttura e le tematiche dell'insegnamento saranno le seguenti:

- Introduzione, Motivazioni, Architetture di Agenti Intelligenti;
- Risoluzione di Problemi e Cenni di Sistemi con Vincoli;
- Pianificazione;
- Trattamento dell'Incertezza e Ragionamento Probabilistico;
- Teoria delle decisioni;
- Cenni di Apprendimento Automatico.

**Modalita' di esame :**

Lo studente deve superare un esame scritto. Inoltre lo studente deve sviluppare un progetto applicativo concordato con il docente.

**Criteri di valutazione :**

La valutazione dello studente si basa su una verifica dell'apprendimento dei concetti di base introdotti durante il corso e sulla capacita' di analisi dello studente.

La valutazione del progetto considera la capacita', da parte dello studente, di individuare un caso di studio adeguato e di svolgere in modo autonomo un'attivita' di progettazione e realizzazione qualitativamente appropriata.

**Testi di riferimento :**

Stuart Russell, Peter Norvig, *Artificial Intelligence: A modern approach.* : Prentice Hall, 2010

**Eventuali indicazioni sui materiali di studio :**

Materiale aggiuntivo sara' disponibile sul sito e-learning del corso.

---

## LINGUA INGLESE B2 (ABILITÀ PRODUTTIVE)

(Titolare: Prof. GILBERTO FILE')

**Periodo:** I anno, annuale

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** ; 3,00 CFU

**Contenuti :**

Acquisizione di competenze linguistiche in lingua veicolare (B2 parlata)

**Testi di riferimento :**

CONTENUTO NON PRESENTE

---

## LINGUAGGI PER IL GLOBAL COMPUTING

(Titolare: Prof. PAOLO BALDAN)

**Periodo:** I anno, 2 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Il corso non ha propedeuticit .

**Conoscenze e abilit  da acquisire :**

L'enorme diffusione dei sistemi concorrenti, distribuiti e mobili rende inadeguati i paradigmi di specifica e programmazione classici ed apre sfide complesse e affascinanti. Appare necessario un ripensamento, che parta dalle stesse fondamenta e che adotti un approccio rigoroso, formale, disciplinato. Il corso si propone di avvicinare lo studente a tematiche di interesse in questo ambito, utilizzando come strumenti sistemi di tipi, calcoli di processo e in generale linguaggi di modellazione. Parte da argomenti fondazionali oramai classici (come il Calculus of Communicating Systems ed il pi-calculus) e giunge ad illustrare alcuni argomenti di punta della ricerca nell'area. Vengono discussi alcuni linguaggi che traducono in pratica gli sviluppi teorici descritti, quali linguaggi evoluti per la concorrenza (Google Go, Erlang, Clojure), linguaggi di orchestrazione (ORC) e linguaggi per programmazione service oriented (Jolie).

**Attivit  di apprendimento previste e metodologie di insegnamento :**

Lezioni in classe e uso di strumenti di verifica automatica.

**Contenuti :**

La struttura e le tematiche del corso saranno le seguenti:

- Introduzione alla concorrenza e mobilit : dagli automi ai sistemi reattivi e concorrenti.

- Calculus of Communicating Systems (CCS), un linguaggio minimale per la descrizione di sistemi concorrenti. Equivalenza di processi: Sistemi di transizione e bisimulazione.

- Logica di Hennessy-Milner e strumenti per la verifica. Mutua esclusione, deadlock, fairness. Propriet  di safety e liveness. Verifica con strumenti automatici. Il Concurrency Workbench e CAAL.

- Dai linguaggi di specifica ai linguaggi di programmazione: linguaggi avanzati per la concorrenza (Google Go e channel-based concurrency, Erlang e modello ad attori, Clojure e functional concurrency)

- Linguaggi di orchestrazione (ORC) e linguaggi per programmazione orientata ai servizi (Jolie).

**Modalit  di esame :**

Esercizi in classe, soluzione e discussione orale di esercizi avanzati, presentazione di un tema scelto dallo studente. Tra le opzioni ci sar  anche la realizzazione di un piccolo progetto.

**Criteri di valutazione :**

Lo studente   valutato rispetto alla sua capacit  di risolvere semplici esercizi, verificando cos  l'acquisizione di nozioni e tecniche discusse durante il corso. Alcuni esercizi avanzati sono finalizzati a verificare la capacit  di mettere a frutto quanto appreso per la soluzione di problemi nuovi. La presentazione verifica l'abilit  dello studente di approfondire, autonomamente, tematiche di ricerca nell'area di interesse per il corso, e di esporre in modo efficace quanto appreso.

**Testi di riferimento :**

Aceto, Luca, Reactive systems modelling, specification and verification Luca Aceto ... [et al.]. Cambridge: Cambridge University Press, 2007

**Eventuali indicazioni sui materiali di studio :**

Il libro di testo   complementato con articoli di ricerca e altre risorse disponibili online.

Pagina web: <http://www.math.unipd.it/~aldan/Global>

**METODI E MODELLI PER L'OTTIMIZZAZIONE COMBINATORIA**

(Titolare: Dott. LUIGI DE GIOVANNI) - Mutuato da: Laurea magistrale in Informatica (Ord. 2014)

**Periodo:** Il anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 32A+4E+12L; 6,00 CFU

**Prerequisiti :**

Elementi di ricerca operativa, elementi di programmazione lineare, elementi di base di programmazione.

**Conoscenze e abilit  da acquisire :**

Uso di metodologie avanzate di supporto alle decisioni per la modellazione e la soluzione di problemi di ottimizzazione combinatoria. Il corso intende fornire strumenti matematici e algoritmici per la soluzione di problemi pratici di ottimizzazione con l'utilizzo dei pacchetti software e delle librerie di ottimizzazione pi  diffusi.

**Attivit  di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali, esercitazioni in laboratorio e discussione di esempi notevoli. Le esercitazioni in laboratorio consistono nell'implementazione di algoritmi di ottimizzazione combinatoria sia esatti (con l'uso di librerie di programmazione lineare intera) sia euristici).

**Contenuti :**

1. Approfondimenti e applicazioni di Programmazione Lineare e dualit  : metodo del semplice primale-duale, tecniche di generazione di colonne, applicazioni a problemi di ottimizzazione su grafo.
2. Metodi avanzati di Programmazione Lineare Intera (PLI): Branch & Bound e tecniche di rilassamento, formulazioni alternative di modelli PLI, metodo dei piani di taglio e tecniche di Branch & Cut, applicazioni ad esempi notevoli: commesso viaggiatore, problemi di localizzazione, problemi di network design etc.
3. Meta-euristiche di Ottimizzazione Combinatoria: ricerca di vicini e varianti, algoritmi evolutivi.
4. Applicazione di metodi di modellazione e ottimizzazione su grafo.
5. Laboratori: utilizzo di software e librerie di ottimizzazione.

**Modalit  di esame :**

Esame orale sui contenuti del corso. Realizzazione facoltativa di un progetto individuale sulla soluzione di un problema, reale o realistico, di ottimizzazione combinatoria (definizione del problema, modellazione, applicazione di un metodo di soluzione esatto e/o euristico).

**Criteri di valutazione :**

L'esame verifica il livello di apprendimento degli argomenti svolti e la capacità dello studente di applicarli per la soluzione di problemi reali di ottimizzazione combinatoria.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Dispense fornite dal docente. Articoli scientifici.

## METODI E MODELLI PER L'OTTIMIZZAZIONE COMBINATORIA

(Titolare: Dott. LUIGI DE GIOVANNI)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 32A+4E+12L; 6,00 CFU

**Prerequisiti :**

Elementi di ricerca operativa, elementi di programmazione lineare, elementi di base di programmazione.

**Conoscenze e abilità da acquisire :**

Uso di metodologie avanzate di supporto alle decisioni per la modellazione e la soluzione di problemi di ottimizzazione combinatoria. Il corso intende fornire strumenti matematici e algoritmici per la soluzione di problemi pratici di ottimizzazione con l'utilizzo dei pacchetti software e delle librerie di ottimizzazione più diffuse.

**Attività di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali, esercitazioni in laboratorio e discussione di esempi notevoli. Le esercitazioni in laboratorio consistono nell'implementazione di algoritmi di ottimizzazione combinatoria sia esatti (con l'uso di librerie di programmazione lineare intera) sia euristici).

**Contenuti :**

1. Approfondimenti e applicazioni di Programmazione Lineare e dualità : metodo del simplesso primale-duale, tecniche di generazione di colonne, applicazioni a problemi di ottimizzazione su grafo.
2. Metodi avanzati di Programmazione Lineare Intera (PLI): Branch & Bound e tecniche di rilassamento, formulazioni alternative di modelli PLI, metodo dei piani di taglio e tecniche di Branch & Cut, applicazioni ad esempi notevoli: commesso viaggiatore, problemi di localizzazione, problemi di network design etc.
3. Meta-euristiche di Ottimizzazione Combinatoria: ricerca di vicini e varianti, algoritmi evolutivi.
4. Applicazione di metodi di modellazione e ottimizzazione su grafo.
5. Laboratori: utilizzo di software e librerie di ottimizzazione.

**Modalità di esame :**

Esame orale sui contenuti del corso. Realizzazione facoltativa di un progetto individuale sulla soluzione di un problema, reale o realistico, di ottimizzazione combinatoria (definizione del problema, modellazione, applicazione di un metodo di soluzione esatto e/o euristico).

**Criteri di valutazione :**

L'esame verifica il livello di apprendimento degli argomenti svolti e la capacità dello studente di applicarli per la soluzione di problemi reali di ottimizzazione combinatoria.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Dispense fornite dal docente. Articoli scientifici.

## MOBILE PROGRAMMING E MULTIMEDIA

(Titolare: Dott.ssa OMBRETTA GAGGI)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Sistemi Operativi, Tecnologie Web

**Conoscenze e abilità da acquisire :**

Il corso illustra le principali tecnologie per la codifica, memorizzazione e diffusione di informazioni multimediali, e fornisce le competenze necessarie alla realizzazione di un'applicazione in ambito mobile, dalla progettazione, allo sviluppo fino al deployment e il mantenimento.

**Attività di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali

**Contenuti :**

Introduzione. Sistemi multimediali e ipermediali. I formati dei media. Media e modelli dei dati. Classificazione dei media. Audio, immagini statiche, video. Media statici, media continui, media temporizzati.

- Le immagini. Rappresentazione digitale delle immagini. Risoluzione e profondità di colore. Percezione umana del colore. Modelli per la codifica dei colori. Tecniche di riduzione dei colori. Formati standard per la rappresentazione delle immagini: GIF, PNG, JPEG. Il formato JPEG2000. Le immagini vettoriali.

- L'audio. Rappresentazione digitale delle informazioni audio. Campionamento e quantizzazione. Teorema di Nyquist. Rapporto segnale-rumore. Dimensione dei dati e banda di trasmissione. Formati standard per la codifica dell'audio: WAV, u-Law. I sistemi MIDI.

- Il video. Rappresentazione del segnale video analogico. Standard NTSC e PAL. Il video digitale. Rappresentazione del colore. Sottocampionamento cromatico. Standard H261, H263, MPEG.

- La compressione dei dati. Compressione reversibile e compressione irreversibile. Compressione entropica. Compressione LZW. Compressione dei dati acustici. Elementi di psicoacustica. Bande critiche. Mascheramento spaziale e temporale. Compressione MP3. Compressione JPEG delle immagini. Compressione video. Codifica predittiva. Vettori di movimento. Compressione MPEG.

- Programmazione mobile. Sistemi operativi: diffusione, storia, caratteristiche. Acquisizione di dati tramite sensori. Caratteristiche delle applicazioni per cellulari. Design dell'interfaccia di un'applicazione mobile. Pubblicazione e distribuzione.

- Framework per cross-platform mobile development. Storia, classificazione e caratteristiche. Importanza della scelta di un framework corretto. Consumi energetici.

#### **Modalita' di esame :**

L'esame consiste nello sviluppo di un progetto e una prova orale.

#### **Criteri di valutazione :**

L'esame verifica l'effettivo apprendimento dei concetti esposti durante l'insegnamento. Questo puo' avvenire in forma di discussione orale, oppure applicando quanto appreso nella progettazione e realizzazione di una applicazione per smartphone.

#### **Testi di riferimento :**

Li, Ze-Nian; Drew, Mark S., *Fundamentals of multimedia*. Upper Saddle River: Pearson Prentice Hall, 0  
Josh Clark., *Designing for Touch. : A Book Apart*, 2015

#### **Eventuali indicazioni sui materiali di studio :**

Le slide del corso sono fornite sul sito web del corso

## **PROCESS MINING**

(Titolare: Dott.ssa CHIARA GHIDINI) - Mutuato da: Laurea magistrale in Data Science (Ord. 2017)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

#### **Prerequisiti :**

Basic notions of algorithms, data structures and programming.

#### **Conoscenze e abilita' da acquisire :**

The aim of the course is to introduce and investigate the main methods and concepts that pertain the modeling and analysis of business processes. More in detail, the course will focus on the main modeling languages (BPMN, Petri Nets, and Declare), on the main methodologies for manual modeling and analysis and on the main algorithms for the (semi)automatic modeling and analysis (the so-called process mining). By exploiting concrete software platforms several algorithms will be also investigated in a 'hands-on' fashion on real data.

At the end of the course the students should have a detailed knowledge of the main methods and concepts of business process modeling e mining, of the main metrics used to support the analysis of business processes and of the main algorithms of process mining.

#### **Attivita' di apprendimento previste e metodologie di insegnamento :**

The course consists of lectures. Some practical activities and exercises will require the use of computers.

#### **Contenuti :**

The course will cover the topics listed below:

#### **1. MODELING AND ANALYSIS: THE BPMN PERSPECTIVE**

- Process Identification
- Essential and Advanced Process Modeling in BPMN
- Qualitative Analysis
- Quantitative Analysis
- Process redesign

#### **2. MODELING AND ANALYSIS: THE PETRI NET PERSPECTIVE**

- An introduction to Petri Nets
- Petri nets and colored petri nets
- Simulation based analysis
- Reachability and coverability analysis
- Process modeling and analysis with PN

#### **3. PROCESS MINING**

- Data & Process mining
- Getting the data: the construction of event logs
- An introduction to Process discovery
- Advanced process discovery
- Conformance checking - replay based
- Conformance checking - logic based
- Mining additional perspectives
- Typical use cases, e.g., medical processes

#### **4. DECLARATIVE APPROACHES**

- Declarative approaches and Declare
- Declarative process mining (discovery in Declare) and hybrid approaches

#### **5. PREDICTIVE PROCESS MONITORING**



- Basic Predictive Process Monitoring techniques
- Advanced Predictive Process Monitoring techniques

**Modalità di esame :**

Written exam and project. The project is due and has to be discussed by the end of the course.

**Criteri di valutazione :**

The project work, and the written exam, will be evaluated on the basis of the following criteria: i) student's knowledge of the concepts, methods, and technologies; ii) ability of the student to master the implementation technology; iii) student's capacity for synthesis, clarity, and abstraction, as demonstrated by the written exam and project presentation. The final grade is obtained as the weighted sum of the grades of the written exam (80%) and the project (20%)

**Testi di riferimento :**

Marlon Dumas, Marcello La Rosa, Jan Mendling, Hajo A. Reijers., *Fundamentals of Business Process Management*, : Springer, 2013  
Wil M.P. van der Aalst and Christian Stahl, *Modeling Business Processes: A Petri Net-Oriented Approach*. : Information Systems, 2011  
W. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Berlin: Springer-Verlag, 2011

**Eventuali indicazioni sui materiali di studio :**

Slides, exercises and scientific papers will be provided.

## PROVA FINALE

(Titolare: da definire)

**Periodo:** Il anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** ; 33,00 CFU

## REPERIMENTO DELL'INFORMAZIONE

(Titolare: Prof.ssa MARISTELLA AGOSTI) - Mutuato da:

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

**Conoscenze e abilità da acquisire :**

Competenze critiche e avanzate di ideazione, progettazione, realizzazione e valutazione di sistemi di reperimento dell'informazione (Information Retrieval: IR) e motori di ricerca (search engines).

**Attività di apprendimento previste e metodologie di insegnamento :**

L'insegnamento si basa su lezioni frontali e attività sperimentali.

La frequenza alle lezioni è fortemente consigliata, perché il corso è progettato per studenti che frequentano con costanza le lezioni, seguono le attività sperimentali del corso e realizzano il progetto.

Informazioni relative al diario delle lezioni, al materiale didattico, alle scadenze e modalità d'esame sono rese disponibili agli studenti in Moodle e/o nel Web.

**Contenuti :**

I sistemi e i servizi di reperimento dell'informazione sono sistemi pervasivi che vengono utilizzati ogni giorno in attività di acquisizione di informazioni, di gestione amministrativa, di acquisizione di competenze, di intrattenimento e comunicazione da parte di milioni di utenti esperti e finali. I motori di ricerca e i sistemi di reperimento dell'informazione gestiscono grandi quantità di informazioni, documenti e risorse digitali (big data).

I motori di ricerca (search engines) sono gli strumenti di reperimento dell'informazione più conosciuti e utilizzati per acquisire informazioni su persone, eventi e risultati scientifici. Grazie ai risultati ottenuti e le informazioni acquisite tramite i motori di ricerca si prendono decisioni e si svolgono attività di lavoro e personali.

Sono strumenti di reperimento dell'informazione anche quelli incorporati nei diversi media sociali (social media) che vengono utilizzati ogni giorno per scopi professionali e personali da utenti di tutto il mondo e tutti quei sistemi che permettono di recuperare informazioni dai messaggi di posta elettronica, dai documenti e dagli archivi presenti sui dispositivi di memorizzazione permanente utilizzati nei diversi strumenti di elaborazione delle informazioni che le persone oggi hanno in uso personale.

I contenuti del corso permettono di acquisire competenze per la ideazione, progettazione, realizzazione e valutazione di sistemi di reperimento dell'informazione e motori di ricerca. La presentazione dei contenuti è articolata nelle seguenti parti:

- Elementi introduttivi (quadro d'insieme) per la rappresentazione, gestione e reperimento automatico dell'informazione in formato digitale
- Indicizzazione: strutture dati idonee al reperimento dell'informazione
- Modelli e sistemi per il reperimento dell'informazione
- Valutazione: collezioni sperimentali, misure di efficacia e di efficienza
- Web search: la struttura del Web, interrogazioni e utenti, agenti Web (Web Crawlers, Crawler), Search Engine Optimization (SEO).

**Modalità di esame :**

1. Progetto su un argomento scelto in accordo con il docente e con presentazione seminariale.
2. Esame scritto con domande sui tutti i contenuti del corso.

### 3. Eventuale integrazione orale.

#### **Criteri di valutazione :**

La prova finale serve a verificare l'acquisizione di competenze relative alla capacità di valutare, ideare, progettare, reingegnerizzare, realizzare e gestire sistemi di reperimento dell'informazione e motori di ricerca.

Il progetto serve a verificare in modo sperimentale la capacità di applicare le competenze acquisite.

#### **Testi di riferimento :**

Croft, W. Bruce; Metzler, Donald; Strohman, Trevor, Search engines: information retrieval in practice. Boston: Pearson, 2010

#### **Eventuali indicazioni sui materiali di studio :**

I materiali di studio sono resi disponibili in Moodle e/o nel Web.

Il libro di testo di riferimento è:

Search Engines: Information Retrieval in Practice  
di Donald Metzler, Trevor Strohman, and W. Bruce Croft

Il testo è disponibile e scaricabile liberamente all'URL:

<http://ciir.cs.umass.edu/downloads/SEIRiP.pdf>

Un utile riferimento aggiuntivo è:

Cornelis Joost (Keith) van Rijsbergen. Information Retrieval (2ª Ed). London, Butterworths, 1979

anche questo testo è scaricabile liberamente all'URL: <http://www.dcs.gla.ac.uk/Keith/Preface.html>

Viene resa disponibile in formato elettronico una dispensa sulla valutazione dei sistemi di reperimento dell'informazione a cura di Gianmaria Silvello.

## **SISTEMI CONCORRENTI E DISTRIBUITI**

(Titolare: Prof. TULLIO VARDANEGA)

**Periodo:** 1 anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

#### **Prerequisiti :**

L'insegnamento assume familiarità con l'architettura degli elaboratori tradizionali, con la struttura e le attività dei loro sistemi operativi, particolarmente per quanto attiene a concorrenza, sincronizzazione e gestione dell'I/O, e dei fondamenti delle reti.

Purtuttavia, l'insegnamento non prevede propedeuticità, perché le attività d'aula sono disegnate per aiutare lo studente con lacune a rinfrescare le proprie conoscenze di dominio.

#### **Conoscenze e abilità da acquisire :**

Il corso si propone di:

1) illustrare problematiche e modelli di base e avanzati di concorrenza intesa come parallelismo potenziale, realizzata a software, illustrando e studiando, dal punto di vista del sistema a run-time, le soluzioni proposte da Ada, Java e altri linguaggi dotati di supporto diretto alla concorrenza, come strumenti di sperimentazione e di confronto;

2) analizzare i principi costruttivi e i paradigmi architetturali e realizzativi che stanno alla base dei sistemi distribuiti, nella loro evoluzione da sistemi multiprocessori omogenei a sistemi multicomputer eterogenei, su larga o larghissima scala fino al paradigma del Cloud Computing, anch'essi studiati dal punto di vista del run-time support.

#### **Attività di apprendimento previste e metodologie di insegnamento :**

Il corso si compone di due segmenti principali e di un terzo e ultimo tratto.

- Nel primo segmento si prendono in esame modelli e paradigmi di programmazione concorrente, concentrandosi sulla concorrenza direttamente esprimibile a linguaggio (ossia senza ricorso a librerie esterne), utilizzando Java e Ada come linguaggi di sperimentazione, ma estendendo l'attenzione ad altri paradigmi linguistici.

- Nel secondo segmento si affronta invece l'evoluzione architetturale tecnologica dei sistemi distribuiti, toccando CORBA come paradigma di interconnessione di sistemi eterogenei secondo il modello cliente-servente.

- Nel terzo e ultimo tratto, l'insegnamento culmina con l'illustrazione di approcci particolarmente avanzati come virtualizzazione e cloud computing.

Nell'ambito di ciascun segmento del corso, il docente propone allo studente esercizi da realizzare in proprio in laboratorio per sperimentare direttamente le problematiche progettuali e realizzative e i paradigmi di soluzione illustrati a lezione.

#### **Contenuti :**

Problematiche di concorrenza

- Introduzione storica e metodologica
- Nozione di processo e modalità di sincronizzazione
- Un modello concreto e sue progressive estensioni
- La dimensione temporale
- Cenni sulla virtualizzazione

Problematiche di distribuzione

- Definizioni fondamentali
- Comunicazione e sincronizzazione in distribuito
- Il sistema dei nomi e la nozione di statelessness
- Soluzioni concrete: Java RMI, Ada DSA, CORBA

La frontiera del Cloud Computing

- Origini del paradigma
- Caratteristiche di applicazioni cloud-native

- Le dimensioni della scalabilità .

**Modalità di esame :**

L'esame di profitto consiste nella redazione e nella discussione di una relazione scritta che illustri le problematiche affrontate nello svolgimento di un tema didattico (pratico o teorico) concordato con lo studente, e le soluzioni tecniche adottate per risolverle.

**Criteri di valutazione :**

Il lavoro assegnato in sede d'esame viene accompagnato da intenso dialogo con il docente, che consente allo studente di approfondire le principali problematiche affrontate a lezione e associarle alla realizzazione del progetto assegnato.

La stesura della relazione tecnica mette alla prova la capacità di sintesi e di astrazione dello studente; le argomentazioni tecniche sostenute nella relazione consentono di completare la valutazione il grado di apprendimento complessivo dello studente rispetto ai principali temi della materia.

**Testi di riferimento :**

Alan Burns and Andy Wellings, *Concurrent and Real-Time Programming in Ada*. : Cambridge University Press, 2007

Andrew S Tanenbaum, Maarten van Steen, *Distributed Systems - Principles and paradigms*. : Pearson Education International, 2006

**Eventuali indicazioni sui materiali di studio :**

Il docente pubblica regolarmente tutte le diapositive utilizzate a lezione e anche materiale supplementare utile per l'approfondimento dei temi trattati in aula.

---

## SISTEMI REAL-TIME

(Titolare: Prof. TULLIO VARDANEGA)

**Periodo:** 1 anno, 2 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 40A+8E; 6,00 CFU

**Prerequisiti :**

L'insegnamento assume familiarità con l'architettura degli elaboratori tradizionali, con la struttura e le attività dei loro sistemi operativi, particolarmente per quanto attiene a concorrenza, sincronizzazione e gestione dell'I/O. L'insegnamento non prevede propedeuticità.

**Conoscenze e abilità da acquisire :**

Il corso si propone di esaminare la struttura dei sistemi software embedded soggetti a vincoli temporali, con l'obiettivo di evidenziarne le caratteristiche che li differenziano dagli altri sistemi di calcolo. Attenzione sarà posta su alcuni paradigmi di progettazione e programmazione di tali sistemi, che ne facilitano l'analisi e la verifica.

**Attività di apprendimento previste e metodologie di insegnamento :**

Il corso esamina la struttura dei sistemi software embedded soggetti a vincoli di tempo reale, illustrando le principali problematiche nella loro progettazione, realizzazione e validazione. In particolare vengono affrontate:

- caratterizzazione architetturale (livello hardware, software, e sistema)
- controllo e gestione del tempo e delle interfacce hardware
- progettazione e programmazione di software real-time
- tecniche e approcci per la modellazione e l'analisi di sistemi real-time
- problematiche di verifica e validazione.

Nell'ambito del corso, il docente propone allo studente esercizi da realizzare in proprio in laboratorio per sperimentare direttamente le problematiche progettuali e realizzative e i paradigmi di soluzione illustrati a lezione, oltre a familiarizzare gli studenti con i più recenti sviluppi della teoria real-time intorno a tematiche di particolare interesse.

**Contenuti :**

- Introduzione: cenni storici e visione architetturale
- Cenni sulla affidabilità e la tolleranza ai guasti
- Il problema dell'ordinamento, tassonomia di algoritmi
- Politiche di sincronizzazione nella gestione delle risorse condivise
- Problematiche di sistema: una visione d'insieme della pila tecnologica
- Estensione ai sistemi distribuiti
- Estensione ai sistemi multiprocessore

**Modalità di esame :**

L'esame si svolge in una di due modalità a scelta dello studente. Una modalità richiede la redazione e la presentazione di una relazione tecnica sulle problematiche incontrate nell'adattamento a principi di progettazione e programmazione real-time di un piccolo sistema concorrente e distribuito individuato congiuntamente dallo studente e dal docente. L'altra modalità prevede lo studio critico e la presentazione di un lavoro di ricerca recente, che sviluppa qualcuno dei temi toccati in aula, scelto dallo studente tra un insieme di lavori individuati dal docente.

**Criteri di valutazione :**

Lo sviluppo della prova d'esame scelta dallo studente, indipendentemente dalle sue specifiche modalità, viene accompagnato da intenso dialogo con il docente, che consente allo studente di approfondire le principali problematiche affrontate a lezione e associate alla realizzazione del progetto. La presentazione e discussione da effettuare in sede d'esame consente di completare la valutazione il grado di apprendimento complessivo dello studente rispetto ai principali temi della materia.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Il docente pubblica regolarmente tutte le diapositive utilizzate a lezione e anche materiale supplementare utile per l'approfondimento dei temi trattati in aula.

---

## START-UP IN ICT

(Titolare: Prof. RUGGERO FREZZA) - Mutuato da: Laurea magistrale in Informatica (Ord. 2014)

**Periodo:** Il anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Nessuno.

**Conoscenze e abilità da acquisire :**

Le conoscenze e abilità sulle quali si focalizza il corso sono (1) l'individuazione di vettori di innovazione e la comprensione del loro potenziale di impatto sul business; (2) i processi dell'innovazione, attraverso le fasi principali di start-up e scale-up, con l'analisi del ruolo che in esse giocano investitori e finanziatori; (3) attività di laboratorio per attraversare, in breve, i passaggi da (1) verso l'avvio di (2).

**Attività di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali, visite a realtà significative, testimonianze in aula, attività di laboratorio.

**Contenuti :**

- Introduzione all'innovazione e all'imprenditorialità.

- Vettori di innovazione: Internet of things. Blockchain. Augmented Reality. Cyber physical systems (edge, fog, cloud). Esempi di applicazione al business

- I processi dell'innovazione: come si deve governare la fase di start-up, la più rischiosa e intangibile? Governance della fase di start-up. Come si calcola il valore di una impresa in fase di start up?

- Investitori istituzionali: business angels e venture capital. Come funzionano, quali sono i loro obiettivi, cosa cercano?

- Il sostegno pubblico alla fase di start up, normativa e strumenti disponibili. Perché il pubblico sostiene l'innovazione in fase di start up? Programmi di sostegno in Italia e all'estero.

- Gestione dell'equity di una start up. L'equity di una start up serve a motivare il team, a raccogliere le risorse finanziarie necessarie ad aprire collaborazioni.

- Illustrazione di temi di progetti di gruppo (casi di studio) e modalità di svolgimento. Approfondimenti ed esercizi sui casi di studio. Laboratorio su progetti di gruppo (anche con visite presso aziende o stakeholder dell'innovazione)

- I processi dell'innovazione: quando termina la fase di start up? Quali sono gli obiettivi della fase successiva, quella di scale up?

- I processi dell'innovazione: la fase di scale up. Come si struttura l'organizzazione nella fase di scale up? (testimonial, portare manager di imprese informatiche)

- I processi dell'innovazione: come finanziare la fase di scale up; il sostegno pubblico alla fase di scale up. (testimonial dal mondo del debito e delle banche)

**Modalità di esame :**

Gli studenti si costituiranno in piccoli gruppi (orientativamente 3 persone gruppo), per lavorare sull'individuazione di un vettore di innovazione di interesse, all'interno di un insieme di temi indicato dal docente, e su di esso costruire una proposta di valore da sottoporre al docente e ad esperti esterni per un giudizio finale. L'attività inizia a metà del programma e si svolge, anche in aula, in modo da favorire l'interazione con il docente.

**Criteri di valutazione :**

Valutazione dei docenti e degli esperti esterni.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

L'insegnamento non adotta un testo di riferimento. Le attività frontali saranno accompagnate da diapositive e da riferimento a materiale di approfondimento nel pubblico dominio.

## START-UP IN ICT

(Titolare: Prof. RUGGERO FREZZA)

**Periodo:** I anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

**Prerequisiti :**

Nessuno.

**Conoscenze e abilità da acquisire :**

Le conoscenze e abilità sulle quali si focalizza il corso sono (1) l'individuazione di vettori di innovazione e la comprensione del loro potenziale di impatto sul business; (2) i processi dell'innovazione, attraverso le fasi principali di start-up e scale-up, con l'analisi del ruolo che in esse giocano investitori e finanziatori; (3) attività di laboratorio per attraversare, in breve, i passaggi da (1) verso l'avvio di (2).

**Attività di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali, visite a realtà significative, testimonianze in aula, attività di laboratorio.

**Contenuti :**

- Introduzione all'innovazione e all'imprenditorialità.

- Vettori di innovazione: Internet of things. Blockchain. Augmented Reality. Cyber physical systems (edge, fog, cloud). Esempi di applicazione al business

- I processi dell'innovazione: come si deve governare la fase di start-up, la più rischiosa e intangibile? Governance della fase di start-up.

Come si calcola il valore di una impresa in fase di start up?

- Investitori istituzionali: business angels e venture capital. Come funzionano, quali sono i loro obiettivi, cosa cercano?

- Il sostegno pubblico alla fase di start up, normativa e strumenti disponibili. perché il pubblico sostiene l'innovazione in fase di start up? Programmi di sostegno in Italia e all'estero.

- Gestione dell'equity di una start up. L'equity di una start up serve a motivare il team, a raccogliere le risorse finanziarie necessarie ad aprire collaborazioni.

- Illustrazione di temi di progetti di gruppo (casi di studio) e modalità di svolgimento. Approfondimenti ed esercizi sui casi di studio. Laboratorio su progetti di gruppo (anche con visite presso aziende o stakeholder dell'innovazione)

- I processi dell'innovazione: quando termina la fase di start up? Quali sono gli obiettivi della fase successiva, quella di scale up?

- I processi dell'innovazione: la fase di scale up. Come si struttura l'organizzazione nella fase di scale up? (testimonial, portare manager di imprese informatiche)

- I processi dell'innovazione: come finanziare la fase di scale up; il sostegno pubblico alla fase di scale up. (testimonial dal mondo del debito e delle banche)

#### **Modalità di esame :**

Gli studenti si costituiranno in piccoli gruppi (orientativamente 3 persone gruppo), per lavorare sull'individuazione di un vettore di innovazione di interesse, all'interno di un insieme di temi indicato dal docente, e su di esso costruire una proposta di valore da sottoporre al docente e ad esperti esterni per un giudizio finale. L'attività inizia a metà del programma e si svolge, anche in aula, in modo da favorire l'interazione con il docente.

#### **Criteri di valutazione :**

Valutazione dei docenti e degli esperti esterni.

#### **Testi di riferimento :**

CONTENUTO NON PRESENTE

#### **Eventuali indicazioni sui materiali di studio :**

L'insegnamento non adotta un testo di riferimento. Le attività frontali saranno accompagnate da diapositive e da riferimento a materiale di approfondimento nel pubblico dominio.

## **STRUCTURAL BIOINFORMATICS**

(Titolare: Dott. DAMIANO PIOVESAN)

**Periodo:** 1 anno, 2 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 32A+16E; 6,00 CFU

#### **Prerequisiti :**

Conoscenze base di algoritmi di ottimizzazione e machine learning. Linguaggio di programmazione: Python.

#### **Conoscenze e abilità da acquisire :**

Il corso intende comunicare le conoscenze di base sulla struttura e funzione della materia vivente nonché i principali metodi computazionali per il loro studio. Inoltre intende permettere allo studente lo svolgimento autonomo di un progetto di ricerca in bioinformatica strutturale, definendo lo stato dell'arte per un problema aperto e un tentativo di risolverlo con lo sviluppo di software che estenda librerie esistenti e la valutazione critica dei risultati ottenuti.

#### **Attività di apprendimento previste e metodologie di insegnamento :**

Il corso si compone di lezioni frontali, esercitazioni pratiche al computer, sviluppo di un progetto e presentazione dello stesso con discussione critica. Le esercitazioni servono per familiarizzare lo studente con le librerie software da usare per un progetto bioinformatico relativo ad un problema attuale diverso per ogni gruppo. La presentazione del progetto richiede una discussione in cui far emergere i punti di forza e debolezza del software implementato.

#### **Contenuti :**

Il corso si compone di due parti:

1) Introduzione alla materia vivente (2 CFU):

- 1.1) Cenni di chimica organica, interazioni deboli ed energetica
- 1.2) Struttura e funzione di DNA e proteine
- 1.3) Lipidi, membrane e trasporto cellulare
- 1.4) Metodi sperimentali per la determinazione strutturale

2) Biochimica computazionale (4 CFU):

- 2.1) Banche dati biologiche
- 2.2) Librerie software e concetti per allineamenti di sequenza e ricerca in banche dati
- 2.3) Relazione sequenza-struttura nelle proteine e classificazione strutturale
- 2.4) Metodi per la predizione della struttura delle proteine da sequenza, l'esperimento CASP
- 2.5) Metodi per la predizione di funzione ed interazioni delle proteine, l'esperimento CAFA
- 2.6) Proteine non globulari, disordine e ripetizioni strutturali

#### **Modalità di esame :**

L'esame si compone di tre parti separate, che devono essere superate tutte: (i valori tra parentesi indicano i pesi per il voto complessivo)

- 1) Test scritto sulle nozioni di biochimica (ca. 30%)
- 2) Progetto software (ca. 40%)
- 3) Presentazione del progetto con valutazione critica (ca. 30%)

### **Criteri di valutazione :**

Viene valutata:

- 1) la comprensione di concetti e gli algoritmi presentati a lezione
- 2) la capacità di applicare le nozioni fornite a lezione su problemi reali
- 3) la capacità critica di saper utilizzare i metodi nei modi più opportuni, scegliendo tra le alternative possibili
- 4) la capacità di sviluppare software riutilizzabile estendendo librerie esistenti
- 5) la capacità espositiva e di discussione critica

### **Testi di riferimento :**

K.C. Mathews, K.E. Van Holde, K.G. Ahern, *Biochimica* (3ª edizione). : Casa Editrice Ambrosiana, 2004

Pascarella, Stefano; Paiardini, Alessandro, *Bioinformatica dalla sequenza alla struttura delle proteine*. Bologna: Zanichelli, 2011

### **Eventuali indicazioni sui materiali di studio :**

Sul sito E-learning vengono resi disponibili molti materiali per il corso. Questi comprendono i lucidi del corso (appena disponibili), le dispense e la letteratura usata per i progetti. Le dispense scaricabili in formato PDF contengono oltre 300 pagine per facilitare lo studio.

---

## **TEORIA DEI TIPI**

(Titolare: Prof.ssa MARIA EMILIA MAIETTI)

**Periodo:** I anno, 2 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 48A; 6,00 CFU

### **Prerequisiti :**

E' caldamente suggerito, ma non strettamente necessario, aver seguito un corso di introduzione alla logica matematica.

### **Conoscenze e abilità da acquisire :**

Lo scopo di questo corso è quello di fornire una introduzione teorica alla teoria dei tipi per poter apprezzare le sue applicazioni in informatica (correttezza dei programmi funzionali e loro verifica in proof-assistent) e in matematica (sviluppo di prove costruttive con eventuale verifica formale al calcolatore ed estrazione del loro contenuto computazionale).

### **Attività di apprendimento previste e metodologie di insegnamento :**

Lezioni frontali e laboratorio.

### **Contenuti :**

Nel corso verranno introdotti i principali concetti di teoria dei tipi al fine di apprezzarne alcune rilevanti applicazioni in ambito matematico, informatico e anche filosofico.

Lo studente verrà introdotto a comprendere i seguenti aspetti della sua multiforme natura:

- 1) La natura computazionale della teoria dei tipi vista come lambda-calcolo tipato a' la Church che la rende un paradigma di un linguaggio di programmazione funzionale, in quanto permette di tipare i programmi con la loro specifica e di verificarne la correttezza in modo interattivo per mezzo di un proof-assistent.
- 2) La natura insiemistica della teoria dei tipi che la rende adatta a formalizzare le dimostrazioni in matematica costruttiva in modo da estrarne il contenuto computazionale.
- 3) La natura predicativa delle costruzioni di tipo dipendente a' la Martin-Löf tramite generazione induttiva. Verranno illustrati esempi di costruzioni non predicative tramite l'utilizzo di paradossi.
- 4) La possibilità di presentare versioni intensionali o estensionali della teoria dei tipi al fine di enucleare importanti proprietà di decidibilità del type-checking utili a costruire un proof-assistent affidabile che permetta di formalizzare le prove matematiche in modo interattivo.

Il corso includerà una parte in laboratorio che introdurrà gli studenti all'utilizzo di un proof-assistent (il francese Coq o il bolognese Matita o lo svedese Agda).

### **Modalità di esame :**

L'accertamento di profitto avverrà con una prova orale dopo il completamento di esercitazioni personali da parte dello studente.

### **Criteri di valutazione :**

L'esame intende valutare le conoscenze acquisite dallo studente sui temi del corso e le sue capacità di svolgere del lavoro autonomo su di essi.

### **Testi di riferimento :**

Bengt Nordström, Kent Petersson, Jan M. Smith, *Programming in Martin-Loef's Type Theory*. : Oxford University Press, 1990

P. Martin-Löf, *Intuitionistic type theory. Notes by G. Sambin of a series of lectures given in Padua, June 1980*. : Bibliopolis, 1984

### **Eventuali indicazioni sui materiali di studio :**

Appunti forniti dal docente.

---

## **TIROCINIO FORMATIVO E DI ORIENTAMENTO**

(Titolare: Dott. DAVIDE BRESOLIN)

**Periodo:** Il anno, 2 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** ; 3,00 CFU

### **Contenuti :**

Attività di tirocinio aziendale, coerente con il piano di studi o con gli obiettivi della tesi. Internship aziendali, attività di tutorato, workshop di interesse informatico e assimilabili.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

## VERIFICA DEL SOFTWARE

---

(Titolare: Prof. FRANCESCO RANZATO)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

### Prerequisiti :

Conoscenze di base dei linguaggi di programmazione. L'insegnamento non prevede propedeuticità.

### Conoscenze e abilità da acquisire :

Il corso mira ad introdurre metodi e strumenti per la specifica del comportamento, l'analisi statica e la verifica automatica dei programmi e, più in generale, dei sistemi software. In particolare, il corso fornisce una introduzione alla semantica formale dei linguaggi di programmazione ed ai metodi formali per la loro analisi statica e verifica.

### Attività di apprendimento previste e metodologie di insegnamento :

L'insegnamento prevede lezioni frontali e la risoluzione in modo indipendente a casa di vari esercizi e/o lo sviluppo di un progetto di verifica del software.

### Contenuti :

- Semantica dei programmi: Modellazione del comportamento (in particolare il comportamento input/output) dei programmi mediante la teoria dell'ordinamento e dei punti fissi. (cf. [https://en.wikipedia.org/wiki/Semantics\\_\(computer\\_science\)](https://en.wikipedia.org/wiki/Semantics_(computer_science)) )

- Analisi statica e verifica di programmi mediante interpretazione astratta: L'interpretazione astratta è una nota tecnica basata su una approssimazione della semantica dei programmi che permette di specificare le proprietà dei programmi deducibili mediante analisi statica e di provarne la correttezza. (cf. [https://en.wikipedia.org/wiki/Abstract\\_interpretation](https://en.wikipedia.org/wiki/Abstract_interpretation) )

- Analisi statica dataflow di programmi: tecnica per dedurre staticamente informazioni sull'insieme dei possibili valori delle variabili nei vari punti del programma. Un grafo di flusso del controllo è utilizzato per determinare le parti di un programma a cui un particolare valore assegnato ad una variabile potrebbe propagarsi. Le informazioni raccolte sono spesso utilizzate dai compilatori (come gcc e javac) per ottimizzare un programma. (cf. [https://en.wikipedia.org/wiki/Data-flow\\_analysis](https://en.wikipedia.org/wiki/Data-flow_analysis) )

- Strumenti di verifica del software: ad esempio, Clousot (Microsoft, USA), Interproc (INRIA, Francia), Jandom (Università di Pescara) (cf. [https://en.wikipedia.org/wiki/List\\_of\\_tools\\_for\\_static\\_code\\_analysis](https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis) )

### Modalità di esame :

Esame orale e/o progetto software, possibilmente suddivisi in parti distinte.

### Criteri di valutazione :

L'esame orale verte su vari esercizi che lo studente deve svolgere in modo indipendente a casa. Il progetto di laboratorio verte su qualche tool di verifica del software.

### Testi di riferimento :

H. Riis Nielson, F. Nielson, *Semantics with Applications: A Formal Introduction*. : Wiley, 1992

Antoine Miné, Tutorial on static inference of numeric invariants by abstract interpretation. : Now, *The Essence of Knowledge*, 2017

### Eventuali indicazioni sui materiali di studio :

Le slide utilizzate a lezione verranno distribuite.

## WEB INFORMATION MANAGEMENT

---

(Titolare: Prof. MASSIMO MARCHIORI)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 48A; 6,00 CFU

### Prerequisiti :

E' opportuno avere familiarità con gli elementi di base del web, così come forniti nel corso di "Tecnologie Web", in particolare HTML, CSS, XML, XSLT.

### Conoscenze e abilità da acquisire :

L'obiettivo principale del corso è quello di introdurre alcune tra le principali tecnologie web di livello avanzato relative alla gestione dell'informazione, in modo da avere una visione ad alto livello del web attuale e del suo futuro. Guarderemo oltre la superficie, mostrando le connessioni profonde, e spesso sorprendenti, tra il mondo tecnologico e quello sociale.

### Attività di apprendimento previste e metodologie di insegnamento :

L'insegnamento prevede lezioni frontali, con esempi illustrativi mostrati anche tramite connessione diretta al web.

### Contenuti :

+ Web Usability

Usabilità ed interazione con gli utenti, analisi multi-livello, come costruire un sito web di successo. Confronto tra il mondo desktop e quello mobile.

+ E-commerce

Il caso studio dei siti di e-commerce, specializzazione dell'interazione col cliente.

+ Web Advertisement

La pubblicit  nei siti web, tecniche d'uso ed errori da evitare.

+ Web Search

Web Site Search, Search Engine Optimization, testo ed ipertesto, il bene ed il male del web, i Social Information Systems.

+ Web Naming

I nomi del web, loro usi ed abusi.

+ Il Web della Conoscenza

Fondamenti del web semantico, rappresentazione della conoscenza, ontologie, semantic querying, syntactic querying, web reasoning, complex systems.

**Modalit  di esame :**

Lo studente deve superare uno scritto, e consegnare un progetto. Sopra una certa soglia minima di punteggio lo studente pu  opzionalmente richiedere un ulteriore esame orale.

**Criteri di valutazione :**

Il criterio di valutazione principale   la comprensione delle tecnologie web mostrate durante il corso. Questo significa quindi conoscere il funzionamento, i punti deboli ed i punti di forza delle tecnologie, la loro interazione nel contesto.

**Testi di riferimento :**

CONTENUTO NON PRESENTE

**Eventuali indicazioni sui materiali di studio :**

Il materiale di studio per l'esame   fornito tramite il sito web del corso attraverso risorse online.

---

## WIRELESS NETWORKS

(Titolare: Prof. CLAUDIO ENRICO PALAZZI) - Mutuato da: Laurea magistrale in Informatica (Ord. 2014)

**Periodo:** Il anno, 1 semestre

**Indirizzo formativo:** Corsi comuni

**Tipologie didattiche:** 40A+8L; 6,00 CFU

**Prerequisiti :**

Reti di Calcolatori

**Conoscenze e abilita' da acquisire :**

Questo corso offre una panoramica delle problematiche inerenti sistemi e servizi basati su reti wireless. A questo scopo, sono analizzati i principali problemi e soluzioni protocollari disponibili per ambienti wireless. Inoltre, sono discussi la terminologia, il funzionamento e le possibili alternative allo stato dell'arte nelle comunicazioni wireless. Attraverso l'analisi dei servizi che possono essere offerti su tecnologia wireless, lo studente diventer  consapevole delle possibili evoluzioni ed utilizzi futuri dei sistemi wireless. Infine, il corso si conclude con alcune nozioni utili all'implementazione di un elaborato volto all'analisi e alla progettazione di protocolli/applicazioni wireless.

**Attivit  di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali e la realizzazione di un progetto.

**Contenuti :**

Introduzione alle reti wireless.

Problematiche relative alle reti wireless: perdite per errore e collisione, equita' e ritardi di trasmissione, handoff

Standard MAC: 802.11 a/b/g/n/p/s

Protocolli di trasporto in ambiente wireless: TCP Vegas, TCP Westwood, TCP Hybla, CUBIC.

Reti ad hoc e protocolli di routing: MANET, VANET, DSDV, AODV, DSR.

Applicazioni e servizi su reti mobili.

**Modalit  di esame :**

Gli studenti sono valutati attraverso progetti individuali o di squadra ed attraverso un esame orale sulle tematiche discusse in aula.

**Criteri di valutazione :**

L'esame orale finale e il progetto realizzato consentono di valutare il livello di apprendimento delle nozioni discusse in classe e l'abilit  dello studente nel maneggiare concetti in modo pratico.

**Testi di riferimento :**

William Stallings, Wireless Communications & Networks (2nd Edition). : Prentice Hall, 2005

**Eventuali indicazioni sui materiali di studio :**

Vengono rese disponibili le trasparenze utilizzate in aula.

---

## WIRELESS NETWORKS



(Titolare: Prof. CLAUDIO ENRICO PALAZZI)

**Periodo:** I anno, 1 semestre  
**Indirizzo formativo:** Corsi comuni  
**Tipologie didattiche:** 40A+8L; 6,00 CFU

**Prerequisiti :**

Reti di Calcolatori

**Conoscenze e abilita' da acquisire :**

Questo corso offre una panoramica delle problematiche inerenti sistemi e servizi basati su reti wireless. A questo scopo, sono analizzati i principali problemi e soluzioni protocollari disponibili per ambienti wireless. Inoltre, sono discussi la terminologia, il funzionamento e le possibili alternative allo stato dell'arte nelle comunicazioni wireless. Attraverso l'analisi dei servizi che possono essere offerti su tecnologia wireless, lo studente diventera' consapevole delle possibili evoluzioni ed utilizzi futuri dei sistemi wireless. Infine, il corso si conclude con alcune nozioni utili all'implementazione di un elaborato volto all'analisi e alla progettazione di protocolli/applicazioni wireless.

**Attivita' di apprendimento previste e metodologie di insegnamento :**

L'insegnamento prevede lezioni frontali e la realizzazione di un progetto.

**Contenuti :**

Introduzione alle reti wireless.

Problematiche relative alle reti wireless: perdite per errore e collisione, equita' e ritardi di trasmissione, handoff

Standard MAC: 802.11 a/b/g/n/p/s

Protocolli di trasporto in ambiente wireless: TCP Vegas, TCP Westwood, TCP Hybla, CUBIC.

Reti ad hoc e protocolli di routing: MANET, VANET, DSDV, AODV, DSR.

Applicazioni e servizi su reti mobili.

**Modalita' di esame :**

Gli studenti sono valutati attraverso progetti individuali o di squadra ed attraverso un esame orale sulle tematiche discusse in aula.

**Criteri di valutazione :**

L'esame orale finale e il progetto realizzato consentono di valutare il livello di apprendimento delle nozioni discusse in classe e l'abilita' dello studente nel maneggiare concetti in modo pratico.

**Testi di riferimento :**

William Stallings, *Wireless Communications & Networks (2nd Edition)*. : Prentice Hall, 2005

**Eventuali indicazioni sui materiali di studio :**

Vengono rese disponibili le trasparenze utilizzate in aula.