REVISIONE 1.1 – ottobre 2022:
- aggiornamento responsabile piani di studio area Reti, aggiornamento docenti in tabella insegnamenti
REVISIONE 1.2 - novembre 2022:
- aggiornamento docenti in tabella insegnamenti
REVISIONE 1.3 - dicembre 2022:
- aggiornamento iter APU-Lingua Inglese

GUIDA AL CORSO DI LAUREA IN **INFORMATICA** (d.m. 270) E MANIFESTO DEGLI STUDI

Corso di laurea in Informatica
Dipartimento di Informatica
Università degli Studi di Torino
Via Pessinetto 12 – Torino

Anno Accademico 2022/2023



Prefazione

Gentile Lettrice, gentile Lettore,
questa è la guida del corso di Laurea in Informatica (secondo il DM 270, classe di laurea L31), ed è organizzata in due parti, come da indicazioni ministeriali. La prima parte è la descrizione del Corso di Laurea, questa descrizione costituisce il "Manifesto degli studi – Piano dell'Offerta Formativa sui tre anni per la coorte 2022",[1] mentre la seconda parte descrive tutti gli insegnamenti offerti per il 2022/2023 (per tutte le coorti attive), elencando per ogni insegnamento i docenti di riferimento, il programma di esame, le modalità di esame ed altre informazioni utili. La prima parte è quindi soprattutto rivolta agli studenti e alle studentesse che si immatricolano quest'anno (coorte 2022), che trovano nel Manifesto la descrizione del loro percorso triennale, mentre la seconda parte è di interesse anche per gli studenti e le studentesse delle due coorti precedenti (2020 e 2021) che seguiranno gli insegnamenti del terzo e secondo anno, rispettivamente. Ricordiamo che per gli studenti e le studentesse immatricolat* negli anni precedenti vale il manifesto degli studi della loro coorte, già pubblicato negli anni precedenti.
Le attività del Corso di Laurea (CdL) di Informatica dell'Università di Torino sono coordinate dal Consiglio di Corso di Studi (CCS), composto dai docenti che insegnano nel CdL e dai rappresentanti degli studenti e delle studentesse, periodicamente eletti. Le attività del CCS sono normate dal "Regolamento del Corso di Laurea in Informatica".

Di seguito alcune note riassuntive, con le principali informazioni di interesse:

1. Tutto quanto è scritto in questa guida (e nei successivi, eventuali aggiornamenti alla guida stessa che verranno pubblicati sul sito del Corso di Laurea) è vincolante per gli studenti e le studentesse della coorte 2022 e per il corso di Laurea e il suo corpo docente. La guida, unitamente ai regolamenti e alle scadenze di Ateneo per tasse, piano carriera ed esame di laurea, costituisce la base del patto "studenti-Università": conoscerla e averne compreso i contenuti è un obbligo dello studente/della studentessa.
2. Il Corso non è a numero programmato, anche se l'accesso comporta il superamento di un test di accertamento del possesso dei requisiti minimi da parte dello studente o della studentessa (TARM), erogato su scala nazionale, descritto alla pagina https://di.unito.it/requisitiLT
3. Per tutto quanto riguarda la procedura di immatricolazione, pagamento tasse, supporto agli studi, assegnazione login di Ateneo per l'accesso alle procedure on-line di iscrizione, e molto altro, potete far riferimento alla pagina principale del sito di Ateneo, selezionando poi il

[1] Con il termine 'coorte' ci si riferisce all'insieme degli studenti immatricolati nello stesso anno.

profilo "futura matricola". Tutti i servizi agli studenti e alle studentesse (segreteria studenti, pagamento tasse e altro) sono supportati da un servizio di help-desk. L'help-desk fornisce una risposta scritta e quindi può essere uno strumento utile per risolvere eventuali incomprensioni.

4. Tutte le informazioni non riportate in questa guida (come i programmi dettagliati degli insegnamenti, gli orari e la localizzazione delle aule) sono reperibili dal sito Internet del Corso di Laurea, all'indirizzo http://laurea.educ.di.unito.it/.

5. Il corso di Laurea in Informatica è uno dei corsi di Laurea della Scuola di Scienze della Natura dell'Università di Torino.

6. La durata della Laurea è di tre anni accademici. I primi due anni sono uguali per tutti gli studenti e le studentesse: gli insegnamenti dei primi due anni sono suddivisi fra corso A, corso B e corso C e gli insegnamenti di laboratorio sono ulteriormente suddivisi, per permettere un adeguato rapporto docente/student*. Il terzo anno è invece articolato in tre diverse aree tematiche di approfondimento, al fine di permettere allo studente/alla studentessa una maggiore personalizzazione degli studi.

7. La Laurea in Informatica ha come naturale prosecuzione la Laurea Magistrale in Informatica, della durata di due anni accademici, laurea che, al momento, è strutturata in tre indirizzi.

8. La Laurea dà anche accesso ai corsi di master di primo livello. I master hanno un obiettivo più spiccatamente professionalizzante e vengono organizzati in collaborazione con aziende del settore e altri enti formativi, e la loro offerta varia di anno in anno.

9. Il termine "Corso di Laurea" è usato in questa Guida per riferirsi al Corso di Laurea in Informatica (triennale) della classe L31 (Scienze e Tecnologie Informatiche) attivato dall'Università di Torino e con sede presso il Dipartimento di Informatica.

10. L'Università di Torino adotta una procedura telematica per l'iscrizione agli esami e la registrazione dei voti conseguiti (con conseguente abolizione del "libretto degli esami" in forma cartacea). Per ogni esame, nonché per la registrazione del superamento dello stage, è obbligatoria l'iscrizione a seguito della valutazione dell'insegnamento da parte dello studente o della studentessa.

11. La valutazione degli insegnamenti avviene con procedura telematica quando si aprono le cosiddette "finestre di valutazione", tipicamente fra la fine del semestre e la fine degli appelli di esame. La valutazione di un insegnamento è condizione necessaria per l'iscrizione all'esame dell'insegnamento stesso. Gli studenti e le studentesse possono poi esprimere una loro valutazione sulle modalità di esame e sulla congruenza fra esame e insegnamento nelle valutazioni dell'anno successivo.

12. Potete trovare la documentazione completa del Corso di Studi, così come caricata annualmente sul sito del Ministero dell'Istruzione, l'Università e la Ricerca (MIUR), alla voce Scheda Unica Ateneo, Laurea L31, Università di Torino, sul sito MIUR www.universitaly.it

In ultimo, un consiglio per gli studenti e le studentesse che si iscrivono *a tempo pieno*. La struttura degli insegnamenti e delle prove di verifica è pensata per chi segue regolarmente le lezioni sostenendo gli esami con regolarità, al fine di laurearsi entro i tempi previsti. Seguire questo consiglio renderà il vostro percorso in questo Corso di Laurea più semplice, più interessante e più coinvolgente.

Per ulteriori informazioni, siete invitati a partecipare alla presentazione del Corso di Laurea che si terrà il giorno ***16 settembre 2022,*** alle ore 10.00 con prenotazione su https://www.eventbrite.it/e/biglietti-accoglienza-matricole-cdl-in-informatica-universita-di-torino-331839780717?utm-campaign=social&utm-content=attendeeshare&utm-medium=discovery&utm-term=listing&utm-source=cp&aff=escb. La presentazione sarà in forma ONLINE e verranno forniti i dettagli per connettersi alla riunione con un avviso che verrà pubblicato al link su http://laurea.educ.di.unito.it/con almeno una settimana di anticipo.
La presentazione è aperta anche agli studenti e alle studentesse non ancora immatricolate/i.

Infine, un augurio: spero che gli anni che passerete con noi, impegnat* negli Studi Universitari di Informatica, rappresentino un momento importante, stimolante della vostra crescita personale e professionale. Spero che l'Università non sia per voi solo il luogo degli esami, ma il luogo dell'apprendimento e della crescita personale e sociale. Concentrarsi sull'apprendimento e sulle competenze, e sulle multiformi possibilità che l'Università vi offrirà per acquisirle, sarà il modo più facile e divertente di portare a termine i vostri Studi con successo.

Buona lettura!

Liliana Ardissono
Professoressa Ordinaria di Informatica,
Presidente del Corso di Laurea e
Laurea Magistrale in Informatica
Tel. 011 6706716
e-mail presccs@educ.di.unito.it

INDICE

PREFAZIONE 2

PRIMA PARTE: MANIFESTO DEGLI STUDI 6

Obiettivi e sbocchi professionali 6
Obiettivi del Corso di Laurea 6
Sbocchi professionali 6
Certificazione di Qualità 7

Organizzazione della didattica 7
Introduzione 8
Requisiti di ammissione 9
Iscrizione come seconda laurea, passaggio o trasferimento. 11
Pre-corsi (corsi di riallineamento) 11
Iscrizione a tempo pieno e a tempo parziale 12
Piano Carriera e responsabili 12
Date dei periodi di lezione 13
Esami 14
Il processo della qualità del Corso di Laurea 14
Riconoscimento Certificazioni di lingua inglese 14
Il tutorato 14
Date e scadenze 14

La Laurea in Informatica 15
Struttura del Corso di Laurea 15
Area tematica di approfondimento Informazione e Conoscenza 18
Area tematica di approfondimento Linguaggi e Sistemi 19
Area tematica di approfondimento Reti e Sistemi Informatici 20
Struttura dell'offerta formativa rispetto al RAD della classe 22

Supporto agli studi 25
Erasmus 26
Biblioteca 26
Supporto on-line agli insegnamenti 27
Aule e laboratori 27
Supporto per studenti e studentesse con disabilità 28
Supporto per studenti e studentesse con DSA (Disturbi Specifici di Apprendimento) 29

Indirizzi utili 31

SECONDA PARTE: PROGRAMMI E DOCENTI DEGLI INSEGNAMENTI PER L'A.A. 2022/23 32
Insegnamenti attivi nel 22/23. 32
Codici per convalide di insegnamenti e competenze. 35
Programmi e altre informazioni per gli insegnamenti attivi nel 22/23 (syllabus degli insegnamenti) 36

Prima parte: Manifesto degli Studi

Obiettivi e sbocchi professionali

Obiettivi del Corso di Laurea

In questi anni il mondo delle imprese, dei servizi e della Pubblica Amministrazione continua a guardare con particolare interesse ai laureati e alle laureate in Informatica e la rapida evoluzione del settore crea sempre nuove opportunità di inserimento degli informatici nel mondo del lavoro.

Il Corso di Laurea fornisce una buona preparazione metodologica di base, coniugata con la comprensione delle più recenti tecnologie in diversi ambiti, dall'impresa al servizio pubblico e alle applicazioni individuali. Obiettivo del corso di laurea è di far acquisire le conoscenze e le competenze necessarie sia per un rapido inserimento nel mondo del lavoro (nel settore delle tecnologie dell'informazione e della comunicazione) sia per permettere al laureato e alla laureata di seguire la rapida evoluzione tecnologica e di adattarsi in modo flessibile a realtà lavorative molto varie.

I laureati e le laureate in Informatica saranno in grado di utilizzare le competenze acquisite nella progettazione, sviluppo e gestione di sistemi informatici; possiederanno inoltre le capacità necessarie per affrontare e analizzare problemi in vari contesti applicativi e per sviluppare sistemi informatici per la loro soluzione.

Oltre alla preparazione nel settore matematico e informatico, integrata dalle competenze in ambito economico e del diritto, il percorso formativo prevede l'acquisizione, da parte dello studente/della studentessa, delle competenze necessarie per lavorare in gruppo e per comprendere e produrre rapporti e documentazione tecnica sia in Italiano che in Inglese.

Sbocchi professionali

I laureati e le laureate in Informatica svolgeranno attività professionale negli ambiti della progettazione, organizzazione e gestione di sistemi informatici, sia in imprese produttrici di hardware o software nelle aree dei sistemi informatici e delle reti, sia nelle imprese che utilizzano le tecnologie dell'informazione nel loro settore specifico (banche, assicurazioni, enti pubblici). Inoltre, le competenze acquisite durante il Corso di Laurea consentono di avviare attività professionali autonome, prospettiva che oggi trova sempre più spazio nel mondo del lavoro.

Il corso prepara alle seguenti **professioni ISTAT**: Tecnici programmatori, Tecnici esperti in applicazioni, Tecnici web, Tecnici gestori di basi di dati, Tecnici gestori di reti e di sistemi telematici.

Il D.P.R. 328/2001 ha istituito nell'Albo professionale degli Ingegneri le Sezioni A e B, in relazione al diverso grado di capacità e competenza acquisita mediante il percorso universitario (magistrale e triennale rispettivamente). Ciascuna sezione è ripartita nei seguenti Settori: Civile e Ambientale, Industriale e dell'Informazione. La Laurea in Informatica è una delle lauree previste per l'ammissione all'esame di stato per la professione di **Ingegnere dell'Informazione, Sez. B**.

Certificazione di Qualità

Il GRIN, che è l'organizzazione nazionale dei ricercatori e delle ricercatrici di Informatica, ha istituito, dal 2004 e in intesa con AICA (Associazione Italiana per il Calcolo Automatico), un certificato che attesta la qualità dei Corsi di Laurea e Laurea Magistrale in Informatica italiani. Il nostro Corso di Laurea ha già ricevuto tale certificato nazionale di qualità negli anni scorsi e in particolare nel 2020, l'ultimo al momento attivato.

Informazioni più dettagliate sui criteri e gli obiettivi di questa certificazione si possono ottenere direttamente sul sito GRIN.

Il nostro Corso di Laurea ha partecipato alle attività per l'accreditamento ministeriale dell'Università di Torino, primo grande Ateneo italiano a candidarsi per l'accreditamento. In particolare il nostro corso di laurea è stato uno dei 7 corsi selezionati dai valutatori per le visite di accreditamento che si sono tenute nel novembre 2015.

Organizzazione della didattica

Introduzione

Il ***Corso di Laurea in Informatica*** è un percorso triennale che porta al conseguimento del titolo di Dottore in Informatica; l'iscrizione richiede il possesso di un diploma di scuola secondaria di II grado. In accordo con la riforma degli studi universitari nota come "riforma 270", dal numero del decreto ministeriale che l'ha definita, si tratta di un percorso di studi a sé stante, che trova però il suo naturale completamento nel ***Corso di Laurea Magistrale,*** un percorso di studi di due anni che porta al conseguimento del titolo di Dottore Magistrale in Informatica, oppure nei ***Master di primo livello*** (normalmente della durata di un anno). Il percorso Magistrale è sempre attivo, mentre l'offerta dei Master, normalmente concordata con le aziende, anche in risposta a bandi regionali o europei, non ha una cadenza fissa, una data di inizio e un ambito tematico ricorrente, e quindi le informazioni non sono riportate in questa guida. Il percorso di studi di 5

anni, laurea triennale seguita da magistrale (anche noto come percorso "3+2") può essere completato dal ***Dottorato di Ricerca***, la cui finalità è di preparare i giovani laureati e laureate alla ricerca scientifica, o da ***Master di secondo livello***.

Il *Corso di Laurea* fornisce conoscenze di base ad ampio spettro, accanto a elementi di formazione professionalizzante, al fine di consentire il proseguimento negli studi magistrali, ma anche un inserimento diretto nel mondo del lavoro. Il *Corso di Laurea Magistrale* permette di ottenere una più spiccata specializzazione in importanti settori dell'informatica di base o di proiettarsi verso aspetti interdisciplinari o di ricerca.

Un concetto fondamentale per comprendere l'organizzazione dei Corsi di Laurea è quello di ***Credito Formativo Universitario*** (CFU o semplicemente "credito" nel prosieguo). Le norme di legge prevedono che ogni CFU equivalga a 25 ore di lavoro per lo studente/la studentessa, che comprendono sia le lezioni e le esercitazioni, sia lo studio individuale. Un Corso di Laurea triennale è costituito da 180 CFU, mentre il percorso di una Laurea Magistrale è costituito da 120 CFU e un corso di Master da 60 CFU. Per il Corso di Laurea in Informatica il Regolamento Didattico stabilisce che ogni CFU sia di norma equivalente a 8 ore di lezione/10 ore di esercitazione in aula o in laboratorio + 17/15 ore di studio individuale.

Ad esempio, un insegnamento di 6 crediti, quale Calcolo Matriciale e Ricerca Operativa (primo semestre del primo anno), corrisponderà a 48 ore di lezioni ed esercitazioni, e si assume che allo studente e alla studentessa (che abbiano una buona conoscenza delle tematiche propedeutiche per l'insegnamento, indicati nel seguito come requisiti di base) siano richieste ulteriori 102 ore per lo studio, i ripassi, la preparazione dell'esame, ecc. In tutte le tabelle che seguono, la 'durata' degli insegnamenti è espressa in CFU. È importante ricordare che questi conteggi fanno riferimento a studenti e studentesse che abbiano acquisito nella Scuola Secondaria di II grado una solida preparazione di base, e che ogni studente dovrà trovare, con l'esperienza, il numero di ore e le modalità di studio individuale che meglio gli permettono di acquisire le richieste conoscenze e competenze.

In questa guida, i termini *insegnamento* e *unità didattica* sono usati in modo equivalente.

Requisiti di ammissione

Sono ammessi al Corso di Laurea in Informatica le studentesse e gli studenti in possesso di Diploma di Scuola Media Superiore o titolo ad esso equivalente.

Per frequentare con successo il Corso di Laurea in Informatica non si richiedono competenze informatiche precedenti, ma è indispensabile avere propensione al ragionamento logico e una buona preparazione nelle materie di base della scuola media superiore (in primo luogo la matematica).

Per immatricolarsi al corso di laurea in Informatica, è necessario sostenere il TARM (Test di Accertamento dei Requisiti Minimi) che sarà erogato attraverso il TOLC-S di CISIA. Alla pagina https://www.unito.it/didattica/immatricolazioni-e-iscrizioni sono riportate tutte le informazioni e le raccomandazioni per iscriversi al TOLC.

Il TOLC-S (https://www.cisiaonline.it/area-tematica-tolc-scienze/struttura-della-prova-e-syllabus/) consiste in 50 quesiti suddivisi in 4 sezioni (Matematica di base, Ragionamento e problemi, Comprensione del testo, Scienze di base), cui si aggiunge una sezione di 30 quesiti per la prova della conoscenza della Lingua Inglese, e ha una durata massima di 110 minuti, cui si aggiungono 15 minuti per la sezione relativa al test per la lingua inglese, per un totale di 125 minuti.

Il test ha esito positivo se si ottiene un punteggio di almeno 5 punti nella sezione di Matematica di base (5/20): a chi non raggiunge questa soglia viene assegnato l'OFA (obbligo formativo aggiuntivo); dovrà seguire l'unità didattica di riallineamento di Matematica, messa a disposizione da UNITO su apposita piattaforma e superare il relativo esame. Soglie di superamento dell'esame e modalità di svolgimento verranno pubblicate sul sito del Corso di Laurea.

L'OFA deve essere assolto entro il primo anno di corso.

Non sono previste ulteriori soglie di sufficienza per il superamento del TOLC-S, ma il Corso di Laurea consiglia di seguire le unità di riallineamento a disposizione su piattaforma dedicata nel caso non si raggiunga un punteggio di almeno 2,5/10 nella sezione "Scienze di base". E' altresì messo a disposizione il percorso Passport per rinforzare le competenze trasversali.

Iscrizione come seconda laurea, passaggio o trasferimento.

Gli studenti e le studentesse che intendono iscriversi come seconda laurea, passaggio da altro Corso di Laurea o trasferimenti da altri Atenei devono farsi pre-valutare la carriera inviando l'elenco degli esami sostenuti, completo di codice dell'insegnamento, CFU attribuiti, settore scientifico/disciplinare e riferimento web al programma, alla seguente commissione:

Commissione	E-mail
Commissione Passaggi, Trasferimenti, II lauree e piani di studio	commpassccs@educ.di.unito.it

Per maggiori informazioni consultare il sito di Ateneo nella sezione "passaggi e trasferimenti".

Pre-corsi (corsi di riallineamento)

I pre-corsi sono uno strumento che il Corso di Laurea mette a disposizione degli studenti e delle studentesse che intendano ripassare e re-inquadrare le competenze di matematica e fisica da loro acquisite nelle scuole superiori, i pre-corsi sono associati ai normali insegnamenti: per Analisi Matematica si svolgeranno nelle prime settimane di lezione del secondo semestre, all'interno dell'orario dell'insegnamento di Analisi Matematica (primo anno, secondo semestre). Per Fisica il pre-corso è invece previsto al secondo anno, durante la prima settimana di lezione dell'insegnamento di Fisica.

Iscrizione a tempo pieno e a tempo parziale

La riforma 270 ha introdotto una distinzione importante per quanto riguarda le iscrizioni all'Università: iscritte/i a tempo pieno e iscritte/i a tempo parziale. Gli studenti e le studentesse "a tempo pieno" possono laurearsi nei tre anni previsti, ma devono dedicare tutte le proprie energie (lavorative) allo studio. Considerando i CFU previsti per il Corso di Laurea, si può osservare che in media ciascuno studente e ciascuna studentessa deve acquisire 60 crediti all'anno, che complessivamente corrispondono a 1500 (=60*25) ore di lavoro.
Se però uno studente o una studentessa intende dedicarsi solo parzialmente allo studio (se, ad esempio, ha anche un'attività lavorativa, o se deve dedicare una parte sostanziale della sua giornata a viaggiare fra casa e università), può iscriversi a tempo parziale. Lo svantaggio è che non sarà più possibile laurearsi in tre anni. Si noti che la scelta viene effettuata ogni anno e non ci sono problemi nel cambiarla da un anno all'altro. Per informazioni sulle modalità di pagamento delle tasse universitarie, si veda http://di.unito.it/tasse. Allo stato attuale gli studenti e le studentesse iscritti a tempo pieno potranno registrare un massimo di 80 CFU all'anno, mentre gli studenti e le studentesse iscritti a tempo parziale potranno registrare un massimo di 36 CFU all'anno.

SUGGERIMENTO esami studenti e studentesse a tempo parziale 1 anno:
Programmazione 1 (1° sem.)
Calcolo Matriciale e Ricerca Operativa (1° sem.)
Programmazione 2 (2° sem.)
Architettura degli Elaboratori (2° sem.)
Lingua Inglese 1 (si può seguire al 1 e/o al 2 semestre)

Piano Carriera e responsabili

Gli studenti e le studentesse sono tenuti a presentare il piano carriera, cioè l'elenco di tutti gli insegnamenti del proprio piano di studi, seguendo le istruzioni presenti sulla pagina pubblica e personale del sito dell'Ateneo (www.unito.it): l'applicativo per il piano carriera viene di norma reso disponibile fra ottobre e gennaio. Il piano carriera si compila secondo le modalità e le scadenze descritte sul sito di Ateneo, e in **accordo con**

l'offerta formativa della propria coorte e le regole dei piani carriera ad essa associate, che specificano gli insegnamenti obbligatori e a scelta, e la loro collocazione nei tre anni**.**
La compilazione del Piano Carriera è condizione necessaria per poter sostenere gli esami, anche quelli (tutti obbligatori) del primo anno.

Per ulteriori informazioni e per modifiche dei piani carriera al di fuori delle scelte già previste dall'offerta formativa, si faccia riferimento alla *Commissione passaggi, trasferimenti e piani di studio* del Corso di Laurea di Informatica (commpassccs@educ.di.unito.it). Gli studenti e le studentesse devono contattare la commissione almeno due settimane prima della scadenza per la presentazione del piano carriera. Per ulteriori informazioni sulle aree tematiche, ci si può rivolgere ai relativi docenti responsabili:
- area tematica di approfondimento "Informazione e Conoscenza": Liliana Ardissono (liliana.ardissono@unito.it)
- area tematica di approfondimento "Linguaggi e Sistemi": Viviana Bono (viviana.bono@unito.it)
- area tematica di approfondimento "Reti e Sistemi Informatici": Rossano Gaeta (rossano.gaeta@unito.it)

Date dei periodi di lezione

Per il Corso di Laurea in Informatica, l'Anno Accademico è suddiviso in due periodi didattici (semestri).
Per l'Anno Accademico 2022/2023 le date di inizio e di fine del periodo di lezioni/esami sono pubblicate alla pagina http://di.unito.it/calendario

Gli orari e le aule degli insegnamenti sono disponibili all'indirizzo http://di.unito.it/orarilezioni. La coorte degli studenti e delle studentesse viene divisa in tre per le lezioni frontali (corso A, persone con iniziale del cognome A-D, corso B, persone con iniziale del cognome E-O, corso C, persone con iniziale del cognome P-Z) e in sei per le lezioni di laboratorio (corso A – turno T1, corso A – turno T2, corso B – turno T1, corso B, turno T2, corso C – turno T1, corso C, turno T2). La regola di suddivisione dei laboratori (con eventuali modifiche dovute alla situazione) è descritta sulle pagine web dei singoli insegnamenti (di.unito.it/insegnamenti).

Esami

Per ogni unità didattica oggetto d'esame, sono previsti 5 appelli d'esame all'anno. Le date e gli appelli degli esami saranno disponibili a partire da fine settembre all'indirizzo http://di.unito.it/appelli. Si ricorda che per poter sostenere un esame *ciascuno studente e ciascuna studentessa deve obbligatoriamente iscriversi* allo stesso dalla bacheca degli appelli, disponibile sulla pagina personale sul sito di Ateneo. Le modalità di esame variano da insegnamento a insegnamento, ma rimangono invariate per tutti e cinque gli appelli dell'anno di uno stesso insegnamento. È possibile iscriversi all'esame di un insegnamento solo se:

- si è in regola con il pagamento delle tasse *e*

- l'insegnamento fa parte del proprio libretto per l'anno in corso (o precedenti) *e*
- si è effettuata la valutazione dell'insegnamento stesso (procedura web Edumeter per la raccolta delle opinioni degli studenti e delle studentesse sugli insegnamenti, si veda il punto seguente)

Gli studenti e le studentesse della coorte 2022, quando saranno iscritti/e al secondo o al terzo anno, potranno registrare esami relativi a un anno superiore al primo solo dopo aver registrato esami del primo anno per un numero di crediti complessivo pari ad almeno 21 e avendo superato l'OFA di matematica previsto per il 2022/23.

Al momento sono previste tre sessioni di laurea l'anno, normalmente nei mesi di luglio, novembre, e aprile. Le date per le sessioni di laurea della coorte 2022 saranno rese note entro l'inizio del terzo anno accademico della coorte. Allo stato attuale delle scadenze per l'ammissione alla Laurea Magistrale, l'ultimo esame di laurea utile ai fini dell'iscrizione è quello di novembre.

Si rimanda al regolamento di Ateneo per i dettagli relativi alla consegna della domanda di laurea.

Il processo della qualità del Corso di Laurea

La valutazione della didattica è obbligatoria per tutti gli studenti e tutte le studentesse, così come per il corpo docente. Per poter sostenere l'esame di un insegnamento lo studente e la studentessa devono aver *obbligatoriamente valutato* l'insegnamento stesso. Oggetto della valutazione saranno l'interesse per l'insegnamento, la chiarezza espositiva del/della docente, la regolarità nello svolgimento delle lezioni, le strutture (aule o laboratori) in cui l'insegnamento si è tenuto. Tale procedura, supportata dall'applicativo Edumeter, è obbligatoria anche per gli studenti e le studentesse non frequentanti (che avranno un diverso insieme di domande) e per i/le docenti (che potranno esprimere un parere sulle strutture e sulla preparazione e interesse degli studenti e delle studentesse). Inoltre, una volta l'anno, gli studenti e le studentesse devono valutare gli insegnamenti che hanno sostenuto nell'anno precedente rispetto alle modalità di esame e alla corrispondenza fra contenuti dell'esame e programma dell'insegnamento.
Le valutazioni sono raccolte in forma anonima e costituiscono la base per le attività di monitoraggio della didattica erogata dal corso di Laurea. In particolare ogni anno, in autunno, il corso di studi effettua un'attività di monitoraggio annuale delle attività di insegnamento: si analizzano gli eventuali punti di forza e le criticità, si identificano contromisure e le si pianificano nel tempo, inoltre si verifica se le contromisure precedentemente identificate siano state correttamente implementate o siano in fase di svolgimento. Alle attività del monitoraggio partecipano anche i rappresentanti degli studenti o un loro delegato/delegata. Eventuali

problemi con l'applicativo web di valutazione possono essere risolti contattando il servizio di help-desk.

Piano Carriera

Gli studenti/le studentesse sono tenuti a presentare il piano carriera nei tempi e nei modi descritti sul sito web dell'Ateneo. L'applicativo per il piano carriera viene di norma reso disponibile in una finestra temporale fra **ottobre e gennaio (con eventuali modifiche fino a fine maggio)**, con accesso dalla propria pagina personale. La compilazione del Piano Carriera è condizione necessaria per poter sostenere gli esami.

Lo studente o la studentessa dovrà selezionare tutti gli insegnamenti del piano di studi così come indicati da questa guida, con particolare attenzione alla scelta dei cfu liberi (seguendo le note del sistema in fase di compilazione): è responsabilità degli studenti e delle studentesse inserire il piano corretto. Eventuali deviazioni dal piano verranno rilevate solo al momento della iscrizione all'esame di laurea, momento in cui viene effettuato un controllo complessivo della carriera dello studente/della studentessa: piani non conformi porteranno alla mancata iscrizione all'esame di laurea.

Prova finale

La prova finale consiste nella preparazione e presentazione orale di una relazione scritta sull'argomento del tirocinio, che è obbligatorio per tutti gli studenti. Il tirocinio potrà essere svolto presso l'Università, oppure presso un'azienda o un ente esterno, secondo modalità stabilite dal Consiglio del Corso di Studi.

La presentazione orale della relazione avviene alla presenza di una commissione nominata dal Consiglio del Corso di Studi.
Dal momento che ogni studentessa o studente svolge un'attività di tirocinio (vuoi interna al Dipartimento, o esterna, nella forma di stage in convenzione con aziende ed enti pubblici), si è scelto di far vertere la prova finale sull'argomento del tirocinio. La studentessa o lo studente prepara una relazione di tirocinio (elaborato scritto sulle attività di tirocinio: si veda in proposito la pagina Unito) e la presenta nel corso dell'esame di Laurea, davanti ad una commissione di docenti del Corso di studi e del Dipartimento. La commissione, alla luce del lavoro svolto nel tirocinio, della qualità della presentazione scritta e orale, e del risultato degli esami di profitto, assegna il voto finale (in centodecimi) e delibera l'eventuale assegnazione della lode, secondo le precise modalita' definite dal CCL-LM nel Regolamento della prova finale.

Riconoscimento Certificazioni di lingua inglese

Gli studenti e le studentesse possono richiedere il riconoscimento dei certificati B1-B2 (secondo il Common European Framework) per l'esame di Lingua Inglese I compilando l'apposito modulo per il riconoscimento delle APU – Attività Pre-Universitarie (con contributo istanze secondo regolamento Unito: http://di.unito.it/tasse). Il modulo va consegnato on-line sul proprio spazio MyUniTO oppure alla Segreteria Studenti del Polo delle Scienze della Natura, in via S. Croce 6, secondo le scadenze definite dalla Segreteria, di norma a metà giugno di ogni anno.
A seguito di apposita delibera, i cfu riconosciuti saranno direttamente caricati sulla carriera degli studenti e delle studentesse, che non dovranno più sostenere Lingua Inglese I – mfn0590 – 3 CFU, facente parte dei crediti obbligatori del piano carriera del I anno.

Il tutorato

Il tutorato comprende attività di assistenza alle studentesse e agli studenti finalizzate a rendere più efficaci e produttivi gli studi universitari. Tali attività mirano a colmare la distanza tra la scuola secondaria e il mondo universitario nelle prime fasi della carriera scolastica, ma anche a migliorare la qualità dell'apprendimento e fornire consulenza in materia di piani di studio, mobilità internazionale, offerte formative prima e dopo la laurea.

Per le studentesse e gli studenti iscritt* al primo anno del corso di studio sono inoltre previste attività di 'tutorato matricole', in cui studentesse e studenti iscritt* agli anni successivi del corso di laurea mettono a disposizione la loro esperienza per aiutare le colleghe e i colleghi del primo anno a superare eventuali difficoltà di carattere pratico-logistico-organizzativo, oltre ad attività di tutorato disciplinare, in cui - per i corsi su cui le studentesse e gli studenti solitamente incontrano maggiori difficoltà - sono previste esercitazioni di recupero a cura di esperti della materia.

Maggiori informazioni sul portale di Ateneo all'indirizzo: http://www.unito.it/servizi/lo-studio/tutorato

La relazione che descrive le attività di Tutorato del CdS è disponibile all'indirizzo web della Commissione (http://di.unito.it/commtutorato).

Date e scadenze

Per tutte le date, le scadenze e le modalità di immatricolazione e passaggi si faccia riferimento all'apposita sezione del *sito web dell'Ateneo*.

La Laurea in Informatica

Nel seguito viene descritta la struttura del Corso di Laurea e come tale struttura sia declinata negli insegnamenti attivati per l'A.A. 2022/2023. Vengono come prima cosa delineati alcuni aspetti comuni alle tre aree tematiche di approfondimento, vengono poi riportate le tabelle ministeriali (RaD) relative al piano dell'offerta formativa per la classe L-31 e la corrispondente implementazione dell'offerta effettuata dall'Università di Torino, e approvata dal Ministero. Vengono infine descritte le tre aree, con l'elenco degli insegnamenti associati e la loro distinzione in caratterizzanti, affini e liberi, e la loro natura obbligatoria o meno.

Struttura del Corso di Laurea

Il Corso di Laurea in Informatica si articola in tre diverse aree tematiche di approfondimento con i primi due anni in comune, dedicati allo studio di materie propedeutiche per gli insegnamenti successivi. Nel primo e secondo anno sono presenti insegnamenti di matematica, di introduzione alla struttura dei calcolatori, alla programmazione, ai sistemi operativi, agli algoritmi e alle basi di dati, oltre a complementi di economia, diritto e fisica.

Il terzo anno gli studenti e le studentesse dovranno scegliere tra le tre aree tematiche di approfondimento: ***Informazione e Conoscenza*** (*IC*), ***Linguaggi e Sistemi*** (*LS*), ***Reti e Sistemi Informatici*** (*Reti*). Tutte le aree tematiche di approfondimento affiancano alla didattica svolta in aula delle ore di didattica e di esercitazioni svolte in laboratorio e prevedono la formazione anche in ambiti disciplinari di cultura generale come diritto, lavoro di gruppo e conoscenza della lingua inglese. Il percorso formativo viene concluso con lo sviluppo di un progetto, di norma svolto presso un'azienda con la supervisione di un/una tutor aziendale e di un/una tutor accademico, che offre agli studenti e alle studentesse una prima esperienza lavorativa.

Come ricordato in precedenza, per il Corso di Laurea in Informatica è stabilito che ogni *CFU* (che, ricordiamo, corrisponde per legge a 25 ore di lavoro per lo studente/la studentessa) sia di norma equivalente a 8 ore di lezione/esercitazione più 17 ore di studio individuale. Per facilitare l'inserimento nel mondo universitario, si propone un primo anno più leggero, di 57 CFU, seguito da due anni di 60 e 63 CFU. Ricordiamo che, nella definizione del piano carriera occorre rispettare le propedeuticità di frequenza (cioè non è possibile inserire esami di un dato anno se non si sono già inseriti in piano carriera tutti gli esami degli anni precedenti).

Nelle tabelle che compaiono in questa sezione sono riportati solo i dati essenziali degli insegnamenti, che permettono di avere un quadro

complessivo dei diversi anni. Maggiori dettagli (semestre di svolgimento, docenti) si possono trovare nelle parte II di questa guida, mentre le schede di tutti gli insegnamenti sono presenti nell'appendice (syllabus), oltre che, naturalmente, nel sito del Corso di Laurea (http://laurea.educ.di.unito.it/), nella parte relativa alla "Offerta formativa".

La laurea prevede 180 CFU. Nelle tabelle che seguono, che descrivono le unità didattiche, la somma totale dei CFU (primo e secondo anno in comune più terzo anno, specifico per ciascuno dei *percorsi*) è pari a 156. I restanti 24 CFU sono così suddivisi: 9 CFU per lo stage e 3 per la prova finale (si veda il paragrafo successivo) e 12 CFU liberi a scelta dello studente/della studentessa (6 dei quali possono essere dedicati a un'estensione dello stage).

Stage e prova finale

Per completare gli studi, lo studente/la studentessa dovrà effettuare un periodo di stage, la cui durata è proporzionale al numero di crediti, presso un'azienda o un centro di ricerca, allo scopo di avere un contatto diretto con il mondo del lavoro (le informazioni necessarie sono disponibili alla pagina http://di.unito.it/stageccs). Lo stage può essere svolto anche all'interno dell'Università, su specifici progetti all'uopo identificati. Al termine dello stage, lo studente/la studentessa dovrà scrivere una relazione in cui si descriva l'attività svolta e i risultati ottenuti. Sia l'attività di stage che quella di scrittura della relazione verranno effettuate sotto la supervisione di un/una docente del Corso di Laurea e si concluderanno con la prova finale (esame di laurea). Lo stage terminato con successo porta al conseguimento degli associati CFU, da registrarsi sul libretto elettronico come ogni altro esame, mentre la Prova Finale, che include il lavoro di stesura della relazione del lavoro di stage e l'esame di laurea in senso stretto, è valutata in ulteriori 3 CFU.
La stage standard è da 9 CFU (corrispondenti a circa 225 ore in azienda), ma lo studente/la studentessa ha facoltà di aumentare l'impegno dedicato allo stage scegliendo di dedicare 6 dei 12 CFU "liberi – a scelta dello studente'' allo stage, che diventa così da 15 CFU (corrispondenti a circa 375 ore in azienda).
Per poter registrare i CFU corrispondenti allo stage, sia interno che esterno all'Università, la procedura richiede di iscriversi all'appello "stage" sulla piattaforma Esse3 relativa agli appelli d'esame e di inviare una mail al proprio relatore per comunicare l'avvenuta iscrizione. Nel caso di stage da 15 CFU ci si dovrà iscrivere anche all'appello "prolungamento stage". I 3 CFU di Prova Finale si acquisiscono il giorno della laurea, senza necessità di preventiva registrazione.

CFU a scelta libera dello studente

Lo studente/la studentessa può scegliere i 12 CFU liberi all'interno dell'offerta formativa dell'intero Ateneo. La scelta deve essere congruente

con l'obiettivo formativo della classe di laurea in Informatica e pertanto la scelta dei liberi deve essere convalidata da apposita commissione. Per ogni area tematica di approfondimento troverete indicate le scelte per i CFU a scelta libera dello studente/della studentessa che sono automaticamente approvate, senza necessità di convalida. Lo studente/la studentessa può usare i CFU liberi anche allo scopo di acquisire CFU in settori non informatici, per esempio per soddisfare requisiti minimi di accesso ad altre lauree magistrali.

Biennio propedeutico comune

Nelle tabella il numero di CFU indicato come n+x indica che l'insegnamento si compone di n CFU di lezioni e x CFU di lezioni/attività di laboratorio.

I anno

Codice	**Insegnamento**	**CFU**
	Uno a scelta tra i seguenti due: (Chi avesse seguito in precedenza il modulo di "Elementi di Logica Matematica" in modalità on-line come progetto Start@Unito deve selezionare l'insegnamento Elementi di Logica Matematica MAT0139; chi ha mai seguito alcun modulo di tale Progetto deve selezionare cod Inf0291 - Logica. L'appello del corso Start@Unito "Elementi di Logica Matematica MAT0139" e di cod Inf0291 - Logica, è unico e condiviso. Le studentesse e gli studenti sono invitati a iscriversi all'appello registrato sul proprio piano carriera. I crediti acquisiti col superamento della prova d'esame saranno attribuiti conformemente)	
Inf0291	Logica	6
Mat0139	Elementi di Logica Matematica	6
Codice	**Insegnamento**	**CFU**
mfn0570	Analisi Matematica	9
Inf0290	Matematica Discreta	6
mfn0582	Programmazione I	6+3
mfn0585	Programmazione II	6+3
mfn0586	Architettura degli elaboratori	6+3
mfn0588	Calcolo matriciale e ricerca operativa	6
mfn0590	Lingua Inglese I	3
	Totale 1 anno	**57**

II anno

Codice	**Insegnamento**	**CFU**
mfn0597	Algoritmi e strutture dati	6+3
mfn0598	Fisica	6
mfn0600	Elementi di probabilità e statistica	6
mfn0601	Sistemi operativi	6+3+3
mfn0602	Basi di dati	5+4
mfn0603	Linguaggi Formali e Traduttori	6+3
mfn0604	Economia e gestione dell'impresa e diritto	9
	Totale 2 anno	**60**

Area tematica di approfondimento Informazione e Conoscenza

L'area tematica di approfondimento "*Informazione e conoscenza*" (IC) ha lo scopo di preparare laureati in grado di sviluppare e gestire sistemi informatici complessi, con particolare attenzione alle applicazioni Web. L'area tematica di approfondimento pone particolare attenzione allo sviluppo di applicazioni desktop e "mobile" di sistemi autonomi e intelligenti, così come alla progettazione di architetture software robuste e flessibili, all'accesso efficiente ai dati e alla loro analisi intelligente, volta a migliorare la fruizione delle grandi moli di informazioni disponibili online. Fornisce dunque una preparazione di base per rispondere alle esigenze della Società dell'Informazione, adatta alla continuazione degli studi nella Laurea Magistrale, affiancata a una preparazione tecnica volta a preparare lo studente/la studentessa per un rapido inserimento nel mondo del lavoro.

III anno

Codice	**Insegnamento**	**CFU**
mfn0605	Programmazione III	4+2
mfn0606	Sviluppo delle applicazioni software	4+5
mfn0607	Sistemi Intelligenti	6
mfn0608	Interazione Uomo Macchina e Tecnologie WEB	12
mfn1362	Reti I	6
	Crediti liberi a scelta dello studente	**12**
inf0073	Stage	9
inf0074	Prova finale	3
	Totale 3 anno	63

Possibili insegnamenti liberi che rendono il Piano di Studi automaticamente approvato secondo l'area tematica di approfondimento Informazione e Conoscenza.

Codice	**Insegnamento**	**CFU**
mfn0610	Linguaggi e Paradigmi di programmazione **se non si è già scelto mfn1354**	6+3
mfn1354	Linguaggi e Paradigmi di programmazione **se non si è già scelto mfn0610**	6
Inf0090	Calcolabilità e Complessità	6
mfn0617	Economia e gestione dell'innovazione	6
Inf0190	Metodi Formali dell'Informatica – parte A	6
mfn0633	Metodi Formali dell'Informatica	9
mfn0636	Sicurezza	6
mfn0618	Sistemi Informativi	6

inf0003	Logica per l'Informatica	6
inf0004	Storia dell'Informatica	6
Inf0193	Metodologie e Tecnologie Didattiche per l'Informatica	6
inf0072	Prolungamento Stage	6

Area tematica di approfondimento Linguaggi e Sistemi

L'area tematica di approfondimento "*Linguaggi e Sistemi*" (LS) ha lo scopo di preparare laureate e laureati in grado di avere una solida preparazione nell'informatica, sia da un punto di vista fondazionale, sia per lo sviluppo di applicazioni.

Quest'area tematica di approfondimento fornisce numerosi elementi formativi di base nel settore matematico e fisico e, per quanto riguarda il settore informatico, le competenze di base relative alle metodologie di programmazione e ai sistemi di elaborazione. Approfondisce inoltre i linguaggi di programmazione e i sistemi di calcolo.

III anno

Codice	**Insegnamento**	**CFU**
mfn0610	Linguaggi e paradigmi di programmazione	6+3
mfn0633	Metodi Formali dell'Informatica	9
mfn0606	Sviluppo delle applicazioni software	4+5
mfn1362	Reti I	6
mfn0605	Programmazione III	4+2
	Crediti liberi a scelta dello studente	**12**
inf0073	Stage	9
inf0074	Prova finale	3
	Totale 3 anno	63

Possibili insegnamenti liberi che rendono il Piano di Studi automaticamente approvato secondo l'area tematica di approfondimento Linguaggi e Sistemi:

Codice	**Insegnamento**	**CFU**
Inf0090	Calcolabilità e Complessità	6
mfn0617	Economia e gestione dell'innovazione	6
mfn0608	Interazione Uomo Macchina e tecnologie web **se non si è già scelto mfn1353 o inf0002**	12
mfn1353	Interazione Uomo Macchina **se non si è già scelto mfn0608**	6
inf0002	Servizi Web **se non si è già scelto mfn0608**	6

mfn0636	Sicurezza	6
mfn0618	Sistemi Informativi	6
mfn0607	Sistemi intelligenti	6
mfn0634	Tecnologie web **se non si è già scelto inf0002 o mfn0608**	6
inf0003	Logica per l'Informatica	6
inf0004	Storia dell'Informatica	6
Inf0193	Metodologie e Tecnologie Didattiche per l'Informatica	6
inf0072	Prolungamento Stage	6

Area tematica di approfondimento Reti e Sistemi Informatici

L'obiettivo di questa area tematica di approfondimento è quello di formare figure competenti nella progettazione, gestione e controllo di sistemi software. In quanto competenti in sistemi operativi, reti e sicurezza, linguaggi e ambienti di programmazione e basi dati, tali figure saranno in grado di contribuire alla gestione dell'informatizzazione della piccola impresa e della pubblica amministrazione, e di inserirsi attivamente nei progetti delle grandi aziende per lo sviluppo e la gestione di sistemi software complessi.

III anno

Codice	**Insegnamento**	**CFU**
mfn0606	Sviluppo delle applicazioni Software	4+5
mfn0634	Tecnologie web	6
mfn0635	Reti di elaboratori	12
mfn0636	Sicurezza	6
	Insegnamenti a scelta	**6**
Inf0090	Calcolabilità e Complessità (6 CFU)	6
mfn0605	Programmazione III – (6 CFU)	4+2
	Crediti liberi a scelta dello studente	**12**
inf0073	Stage	9
inf0074	Prova finale	3
	Totale 3 anno	63

Possibili insegnamenti liberi che rendono il Piano di Studi automaticamente approvato secondo l'area tematica di approfondimento Reti e Sistemi Informatici:

Codice	Insegnamento	CFU
mfn0610	Linguaggi e Paradigmi di programmazione **se non si è già scelto mfn1354**	6+3
mfn1354	Linguaggi e Paradigmi di programmazione **se non si è già scelto mfn0610**	6
mfn0617	Economia e gestione dell'innovazione	6
mfn1353	Interazione Uomo Macchina	6
Inf0190	Metodi Formali dell'Informatica – parte A	6
mfn0633	Metodi Formali dell'Informatica	9
mfn0618	Sistemi Informativi	6
mfn0607	Sistemi Intelligenti	6
inf0003	Logica per l'Informatica	6
inf0004	Storia dell'Informatica	6
Inf0193	Metodologie e Tecnologie Didattiche per l'Informatica	6
inf0072	Prolungamento Stage	6

Struttura dell'offerta formativa rispetto al RaD della classe

Le tabelle che seguono descrivono l'impostazione dell'offerta formativa del nostro corso di Laurea rispetto alla struttura della classe L-31 così come definita dal Ministero.

Attività di base

ambito disciplinare	settore	CFU		minimo da D.M. per l'ambito
		min	max	
Formazione matematico-fisica	FIS/01 Fisica sperimentale FIS/02 Fisica teorica modelli e metodi matematici FIS/03 Fisica della materia MAT/01 Logica matematica MAT/02 Algebra MAT/03 Geometria MAT/04 Matematiche complementari MAT/05 Analisi matematica MAT/06 Probabilita' e statistica matematica MAT/07 Fisica matematica MAT/08 Analisi numerica MAT/09 Ricerca operativa	18	30	**12**
Formazione informatica di base	INF/01 Informatica ING-INF/05 Sistemi di elaborazione delle informazioni	24	36	**18**
	Minimo di crediti riservati dall'ateneo minimo da D.M. 30:	-		

Attività caratterizzanti

ambito: Discipline Informatiche		CFU	
intervallo di crediti da assegnarsi complessivamente all'ambito **(minimo da D.M. 60)**		69	90
Gruppo	**Settore**	**min**	**max**
C11	INF/01 Informatica	69	90

C12	ING-INF/05 Sistemi di elaborazione delle informazioni	0	18

Minimo di crediti riservati dall'ateneo minimo da D.M. 60:		

Attività affini

ambito: Attività formative affini o integrative		**CFU**	
intervallo di crediti da assegnarsi complessivamente all'attività **(minimo da D.M. 18)**		18	24
A11		6	15
A12		6	15

Descrizione sintetica delle attività affini e integrative

Le attività affini o integrative (TAF C) approfondiscono le conoscenze delle studentesse e degli studenti attraverso lezioni, laboratori ed esercitazioni nei seguenti ambiti:

- scienze fisiche, per approfondire la struttura fisica dei calcolatori e dei circuiti che li compongono;

- scienze matematiche: introduzione ai metodi e alle tecniche dell'analisi matematica, della matematica discreta, della logica, della probabilità, della statistica e dell'analisi numerica sia come strumenti per lo sviluppo e lo studio di modelli di interesse informatico che come basi teoriche per la comprensione delle strutture di calcolo. Prime esperienze di impiego dei dati come passaggio fondamentale per lo sviluppo dei modelli e per la loro validazione;

- scienze giuridiche: diritto, per approfondire tematiche legate alla proprietà intellettuale e alla gestione della privacy;

- scienze economiche e aziendali, per introdurre alla comprensione delle funzioni e processi aziendali, dalla definizione del modello di business alla realizzazione dei percorsi strategici finalizzati alla creazione di valore.

Altre attività

ambito disciplinare		CFU min	CFU max
A scelta dello studente		12	15
Per la prova finale e la lingua straniera (art. 10, comma 5, lettera c)	Per la prova finale	3	9
	Per la conoscenza di almeno una lingua straniera	2	6
Minimo di crediti riservati dall'ateneo alle Attività art. 10, comma 5 lett. c		-	
Ulteriori attività formative (art. 10, comma 5, lettera d)	Ulteriori conoscenze linguistiche	0	6
	Abilità informatiche e telematiche	0	6
	Tirocini formativi e di orientamento	0	15
	Altre conoscenze utili per l'inserimento nel mondo del lavoro	0	6
Minimo di crediti riservati dall'ateneo alle Attività art. 10, comma 5 lett. d		1	
Per stages e tirocini presso imprese, enti pubblici o privati, ordini professionali		0	15

Riepilogo CFU

CFU totali per il conseguimento del titolo	**180**
Range CFU totali del corso	147 - 258

Supporto agli studi

Erasmus

L'Università di Torino ha stabilito accordi bilaterali con un certo numero di Università europee per lo scambio di studenti/studentesse di Informatica nel quadro del progetto Socrates/Erasmus dell'Unione Europea. Gli studenti/le studentesse del Corso di Laurea in Informatica possono quindi concorrere a borse annuali per un soggiorno di studio all'estero, con convalida degli esami colà sostenuti. Dal 16/17, verificata la disponibilità di risorse finanziarie, il Dipartimento ha inoltre previsto un'integrazione di queste borse sulla base dei risultati effettivamente conseguiti.
Gli accordi attualmente esistenti riguardano:
"1 Decembrie 1918" Universitatea Alba Iulia, Romania
Panepistimio Thessalias University, Volos, Grecia
Rijksuniversiteit Groningen, Olanda
Sabaci University, Istanbul, Turchia
Slovak University of Technology in Bratislava, Slovacchia
Technical University of Lodz, Polonia
Technical University of Sofia, Bulgaria
Universidad Autonoma de Barcelona, Spagna
Universidad Autonoma de Madrid, Spagna
Universidad de La Laguna, Isole Canarie-Tenerife Spagna
Universidad de Las Palmas de Gran Canaria, Isole Canarie Spagna
Universidad de Zaragoza, Spagna
Universidad Politecnica de Madrid, Spagna
Universidad Politecnica de Valencia ETSIAMN, Spagna
Universitat Klagenfurt, Austria
Universitatea "Politehnica" din Bucuresti, Romania
Universitatea Tehnica Cluj-Napoca – Technical University of Cluj-Napoca, Romania
Université de Lorraine, Nancy, Francia
Universite de Savoie Mont-Blanc, Chambery Francia
Université des Sciences et Tech. Du Languedoc, Montpellier, Francia
Université du Luxembourg, Lussemburgo
Universite Jean Monnet de Saint-Etienne, Francia
Université Paris Diderot – Paris 7, Parigi, Francia
Université Paris Nord – Paris XIII, Parigi, Francia
Universite Paul Sabatier, Toulouse, Francia
Université Pierre et Marie Curie, Paris 6, Francia (per Erasmus Traineeship)
University of Helsinki, Finlandia
University of Southern Denmark, Danimarca
Western Norway University of Applied Sciences, Bergen, Norvegia
Wroclaw University of Technology, Polonia
Altre informazioni sono disponibili sui siti http://di.unito.it/erasmus e http://di.unito.it/erasmusccs

Biblioteca

Come molte biblioteche universitarie, anche la biblioteca del Dipartimento di Informatica (http://di.unito.it/biblio) contiene collezioni di interesse sia per gli studenti e le studentesse che per i ricercatori.
Per quanto riguarda l'aspetto didattico la biblioteca dispone dei testi di studio relativi ai diversi insegnamenti. Molti insegnamenti, accanto ad un testo fondamentale o a una raccolta di dispense, consigliano altri libri complementari: la biblioteca offre la possibilità di consultare tali testi e di ottenerne il prestito per lo studio individuale.
Il patrimonio documentario è costituito da numerosi testi monografici e riviste specialistiche, in parte accessibili a scaffale aperto, in parte accessibili su richiesta del personale bibliotecario. La biblioteca dispone di una sala studio-consultazione dove è anche possibile effettuare ricerche bibliografiche online dai cataloghi accessibili da due postazioni informatiche.
Per comunicare con la biblioteca: biblioteca@di.unito.it
Per maggiori informazioni e dettagli su orari e servizi, consultare il sito http://di.unito.it/biblioteca.

Supporto on-line agli insegnamenti

Oltre ai libri di testo e a eventuali dispense, i docenti degli insegnamenti utilizzano una piattaforma di e-learning per la distribuzione di ulteriore materiale didattico, per la consegna di eventuali esercizi, la distribuzione di esercizi di autovalutazione, e blog di discussione fra studenti e studentesse dello stesso insegnamento.

Aule e laboratori

Dopo la situazione emergenziale degli scorsi mesi, il Dipartimento e il Corso di Studi stanno organizzando la ripresa delle lezioni in presenza, nel rispetto dei protocolli sanitari e delle disposizioni del Ministero dell'Università e dell'Ateneo.
Per aggiornamenti si prega di consultare frequentemente la pagina di Ateneo
(Disposizioni per chi studia e lavora in UniTo) e l'home page del Corso di Laurea.

In condizioni normali, le lezioni si svolgono nelle aule e nei laboratori situati in Via Pessinetto 12 a Torino, presso la sede del Dipartimento di Informatica dell'Università degli Studi di Torino.

I 5 laboratori didattici informatizzati del Corso di Laurea (vedere la pagina http://di.unito.it/laboratori per dettagli) mettono a disposizione 138 postazioni di lavoro (per circa 300 persone), tutte collegate ad Internet, gestite da server ad elevate prestazioni.
Nello specifico, nei laboratori Turing, Von Neumann, Dijkstra e Babbage si svolgono le lezioni degli insegnamenti di laboratorio, mentre il laboratorio Postel è principalmente dedicato allo sviluppo delle tesi di laurea. Nei periodi in cui i laboratori non sono occupati da lezioni, le postazioni di

lavoro sono a disposizione degli studenti e delle studentesse per esercitazioni individuali e per la preparazione della tesi.
Il laboratorio Turing è accessibile utilizzando le credenziali fornite alle studentesse e agli studenti dall'Ateneo in fase di iscrizione all'Università. Per accedere invece agli altri laboratori servono invece delle credenziali specifiche fornite dal Corso di Laurea. I laboratori sono a disposizione degli studenti e delle studentesse per i soli scopi didattici, e vengono gestiti nella più stretta osservanza delle norme contro la pirateria informatica.

Dopo l'iscrizione all'Università, gli studenti e le studentesse sono invitati/e a richiedere quanto prima le credenziali specifiche del Corso di Laurea, rivolgendosi ai tecnici del Dipartimento di Informatica negli orari fissati (http://di.unito.it/login).

I laboratori sono aperti dal Lunedì al Venerdì dalle ore 8.30 alle ore 19:00.

Supporto per studenti e studentesse con disabilità

L'Università degli Studi di Torino, nella prospettiva di rendere effettivo il diritto allo studio per tutti gli studenti e le studentesse con disabilità, intende garantire l'accesso fisico alle strutture di studio e di ricerca. Esiste un progetto di progressiva eliminazione delle barriere architettoniche che, ogni anno, disponendo di apposita quota di finanziamento, affronta le situazioni che sono state individuate, attraverso un censimento di tutti gli edifici, come maggiormente problematiche e gravi.

Gli studenti e le studentesse con disabilità possono usufruire di agevolazioni relative al pagamento delle tasse in relazione alla percentuale di disabilità.

Gli studenti e le studentesse con disabilità possono, inoltre, richiedere diverse tipologie di servizi:
• accompagnamento presso le strutture universitarie e gli enti di ricerca e assistenza durante i pasti;
• tutorato didattico: aiuto per la compilazione di appunti, il reperimento testi, fotocopie, disbrigo di pratiche burocratiche;
• supporto per la preparazione degli esami (rivolto esclusivamente a studenti e studentesse con particolari disabilità);
• supporto di interpreti della Lingua Italiana dei Segni e di Mediatori alla Comunicazione per gli studenti e studentesse non udenti;
• supporto per la richiesta di prove d'esame individualizzate;
• possibilità di utilizzo dei locali del Settore per attività di studio connesse all'Ateneo e disponibilità di postazioni informatiche accessibili;
• sostegno personalizzato attraverso progetti individuali specifici.

Gli studenti e le studentesse con disabilità possono inoltre accedere ai servizi dell'Ente per il Diritto allo Studio Universitario della Regione Piemonte e del Settore Mobilità Internazionale secondo le modalità individuate dai bandi di concorso.

L'elenco dei Delegati per studenti e studentesse con disabilità delle Scuole e dei Dipartimenti è disponibile sul sito di Ateneo www.unito.it » Servizi per lo Studio » Studenti con disabilità » Delegati

Per informazioni
Direzione Risorse Umane
Settore Integrazione Studenti Disabili
Via Po, 31 – Via Po, 29 (ingresso studenti) – 10124 Torino
Tel. 011.670.4282/4283/4284 – Fax 011.670.4285 – Email: ufficio.disabili@unito.it

Orari:
• apertura dei locali dedicati agli studenti e studentesse per attività di studio e di ricerca, incontri con i propri operatori e utilizzo di postazioni

informatiche accessibili, preferibilmente su prenotazione, da lunedì a giovedì 9-18.45 e venerdì 9-16.30;
• ricevimento studenti e studentesse per colloqui specifici e/o prima accoglienza esclusivamente su appuntamento.

La Scuola di Scienze delle Natura, a cui il corso di Laurea in Informatica afferisce, ha nominato come garante per gli studenti e studentesse con disabilità della propria Scuola la Prof.ssa Anna Capietto, a cui è possibile rivolgersi per informazioni specifiche sui test d'ingresso, sulla modalità d'esame, sui percorsi didattici specifici e particolari. Referente per il nostro Corso di Laurea è la Prof.ssa Cristina Baroglio (supplente Prof. Alessandro Mazzei).

Prof.ssa **Anna Capietto** tel. 0116702914 fax 0116702878 anna.capietto@unito.it Dip. Matematica "Giuseppe Peano" Via Carlo Alberto, 10 – 10123 Torino

Prof.ssa **Cristina Baroglio**, tel. 0116706711, cristina.baroglio@unito.it, Dipartimento di Informatica, Via Pessinetto 12, Torino

Supplente: Prof. **Alessandro Mazzei**, tel. 0116706801, alessandro.mazzei@unito.it, Dipartimento di Informatica, Via Pessinetto 12, Torino

Tutte le informazioni sono reperibili al seguente link:
- studenti e studentesse con disabilità: http://www.unito.it/servizi/lo-studio/studenti-disabili

Supporto per studenti e studentesse con DSA (Disturbi Specifici di Apprendimento)

L'Università di Torino ha uno **sportello dedicato** agli studenti e alle studentesse con DSA. L'Università di Torino, in sintonia con le recenti disposizioni di legge, in specifico la legge 8 ottobre 2010, n. 170, si impegna ad adottare modalità di apprendimento e di valutazione il più possibile flessibili, in grado di **garantire il pieno apprendimento** in termini di conoscenze e competenze anche per gli studenti e le studentesse con DSA. Lo sportello predisposto offre i seguenti servizi:

- interventi di **mediazione** con i/le docenti in vista degli esami orali o scritti;
- **tutorato specifico** (redazione appunti, registrazione lezioni) per le attività didattiche;
- **informazioni** sulle procedure di immatricolazione e sui test d'ingresso;
- incontri individuali di **consulenza didattica**.

L'elenco dei Referenti per gli studenti e le studentesse con DSA è disponibile sul sito di Ateneo www.unito.it » Servizi per lo Studio » Studenti con disabilità » Delegati

In particolare, alla pagina http://www.unito.it/servizi/lo-studio/studenti-con-disabilita/supporto-agli-studenti-con-disabilita-sostenere-gli-esami segnaliamo l'apposito "Modulo di richiesta trattamento individualizzato per esame" che ogni studente/studentessa deve compilare ed inviare via e-mail, con congruo anticipo, al Docente responsabile dell'insegnamento.

Per informazioni
Direzione Risorse Umane
Sportello Dislessia
Via Po, 31 – 10124 Torino
Tel. 011.670.4282/4283/4284 – Fax 011.670.4285 – Email: ufficio.dsa@unito.it
www.unito.it » Servizi per gli studenti » Studenti dislessici

Referente per il nostro Corso di Laurea è la Prof.ssa Cristina Baroglio (supplente Prof. Alessandro Mazzei).

Prof.ssa **Cristina Baroglio**, tel. 0116706711, cristina.baroglio@unito.it, Dipartimento di Informatica, Via Pessinetto 12, Torino
Supplente: Prof. **Alessandro Mazzei**, tel. 0116706801, alessandro.mazzei@unito.it, Dipartimento di Informatica, Via Pessinetto 12, Torino

Tutte le informazioni sono reperibili al seguente link:

http://www.unito.it/servizi/lo-studio/studenti-con-disturbi-specifici-di-apprendimento-dsa

Indirizzi utili

Ufficio del Corso di Laurea e Laurea Magistrale in Informatica
C.so Svizzera 185, Torino (ingresso da Via Pessinetto 12 – I piano)
*da Lunedì a Venerdì solo su appuntamento**
e-mail: informatica@educ.di.unito.it
Telefono: 011 – 670 67 41; 011 – 670 68 25; fax 011 – 751603
**per appuntamento scrivere a: informatica@educ.di.unito.it con almeno 2 giorni lavorativi d'anticipo*

Segreteria Studenti (http://di.unito.it/segreteriaStudenti)
Via Santa Croce, 6 – Torino
Apertura sportelli: Lun e Ven: 9-11; Mar, Mer, Gio: 9-11 e 13.30-15
help desk: https://fire.rettorato.unito.it/helpdesk_ddss/
Telefono: 011 – 670.9900 (numero telefonico unico – selezionare Polo delle Scienze della Natura)

Job Placement (http://di.unito.it/jobplacement)
Via Michelangelo 32 – Torino
Da Lun a Ven: 09.30-12 e 13.30-16
e-mail: jp.scienzedellanatura@unito.it
Telefono: 011 – 670 6215-6216-6218-6219; fax: 011 – 670 6217

Presidente del Consiglio dei Corsi di Laurea e Laurea Magistrale in Informatica:
Prof. ssa Liliana Ardissono *e-mail: presccs@educ.di.unito.it*

Direttore del Dipartimento di Informatica:
Prof. ssa Susanna Donatelli *e-mail: direttore@di.unito.it*

Ufficio Accertamento Economico, Regolamento Tasse, inserimento Fasce di reddito (http://di.unito.it/tasse)
Vicolo Benevello 3/a, Torino
Apertura sportelli: Lun e Ven: 9-11; Mar, Mer, Gio: 9-11 e 13.30-15
help desk: https://fire.rettorato.unito.it/helpdesk_ddss/
Tel. 011 – 670 9902 (solo da lun a ven ore 9.00-12.00); fax 011 – 23 610 73

Ente Diritto allo Studio Universitario (http://www.edisu.piemonte.it)
(borse di studio, posto letto, buoni mensa)
Via Giulia di Barolo, 3/bis – 10124 Torino
Tel. 011 – 652 27 01 E_mail: edisu@edisu-piemonte.it
Lun e Ven: 9-11; Mar, Mer, Gio: 9-11 e 13.30-15

Ufficio Studenti Internazionali (http://di.unito.it/stranieri)
Vicolo Benevello, 3/a (I piano) – 10124 Torino
Lun e Ven: 9-11; Mar, Mer, Gio: 9-11 e 13.30-15
e-mail: internationalstudents@unito.it
Telefono: 011 – 670 4498 oppure 011 – 670 4499

Ufficio Erasmus (http://di.unito.it/erasmus)
Vicolo Benevello 3/A – 10124 Torino
Lun e Ven: 9-11; Mar, Mer, Gio: 9-11 e 13.30-15
Per gli Studenti: internationalexchange@unito.it,
Telefono: 011 – 670 4425

Rappresentanti degli Studenti e delle Studentesse: mail rappstudccs@educ.di.unito.it

Seconda Parte: Programmi e docenti degli Insegnamenti per l'A.A. 2022/23

In questa seconda parte della guida vengono riportati tutti gli insegnamenti attivati nell'anno accademico 22/23 presso il Corso di Laurea di Informatica, di interesse per gli studenti e le studentesse immatricolati nell'anno e per gli studenti e le studentesse delle due coorti precedenti che devono seguire gli insegnamenti del secondo e terzo anno rispettivamente. Sono inoltre riportati gli insegnamenti che verranno attivati nei prossimi anni, nonché l'elenco dei codici per le convalide (per gli studenti e le studentesse in trasferimento).

Insegnamenti attivi nel 22/23.

La tabella che segue riporta tutti gli insegnamenti che saranno attivi nell'anno accademico 2022/23. Nella tabella le colonne "Codice", "Insegnamento" e "SSD" indicano, rispettivamente, il codice di Ateneo, il titolo dell'insegnamento e il suo settore scientifico disciplinare, TAF e Ambito individuano la tipologia dell'attività formativa (di base, caratterizzante, a scelta dello studente/della studentessa, etc.), "CFU" sono i crediti formativi associati all'insegnamento, "docenti" sono i docenti responsabili dell'insegnamento e "semestre" il periodo didattico in cui si tiene l'insegnamento. Tutti gli insegnamenti del primo anno sono triplicati (corso A, corso B e corso C)*, gli insegnamenti del secondo anno sono sdoppiati (corso A e corso B)#, mentre i laboratori sono sestuplicati sul 1 anno e quadruplicati sul 2 anno (turni A1, A2, B1, B2, C1 e C2 oppure T1 (A), T2 (A), T3 (B), T4 (B) – 1 e 3: matricola dispari, 2 e 4: matricola pari). Ricordiamo che analoghe informazioni e aggiornamenti dipendenti dalla situazione sono reperibili sulle pagine web del corso di laurea (da http://laurea.educ.di.unito.it/ seguire il link 'insegnamenti', nella sezione "Per chi studia con noi").

** Corso A: dalla A alla D, Corso B: dalla E alla O, Corso C: dalla P alla Z*
 # Corso A: dalla A alla K, Corso B: dalla L alla Z

Codice principale	INSEGNAMENTO	SSD	TAF	Ambito	CFU	DOCENTI	Semestre	Note
mfn0597	Algoritmi e Strutture Dati	inf/01	B	caratterizzante	9	**A: Horvath** **B: de' Liguoro** **Lab T1: Magro** **Lab T2: Audrito** **Lab T3: Pozzato** **Lab T4: Micalizio**	2 sem	
mfn0570	Analisi Matematica	mat/05	A	di base	9	**A: Barutello, Vita** **B: Boscaggin, Vita** **C: Seiler, Garello**	2 sem	
mfn0586	Architettura degli Elaboratori	inf/01	A	di base	9	**A: Gaeta** **B: Drago** **C: Schifanella C** **Lab A1: Gaeta** **Lab A2: Garetto** **Lab B1: Radicioni** **Lab B2: Garetto** **Lab C1: Lucenteforte** **Lab C2: Schifanella C**	2 sem	
mfn0602	Basi di Dati	inf/01	B	caratterizzante	9	**A: Anselma** **B: Pensa**	2 sem	

						Lab T1: Anselma **Lab T2: Mauro, Vernero** **Lab T3: Pensa** **Lab T4: Vernero**		
inf0090	Calcolabilità e Complessità	inf/01	D	a scelta	6	**Berardi**	1 sem	
mfn0588	Calcolo Matriciale e Ricerca Operativa	mat/09	A	di base	6	**A: Grosso** **B: Aringhieri** **C: Aringhieri, Grosso**	1 sem	
mfn0604	Economia e Gestione dell'Impresa e Diritto	6 CFU secs-p/08 e 3 CFU ius/02	C	affine e integrative	9	**A economia: Pironti + Iaia** **A diritto: Montalcini** **B economia: Pironti + Iaia** **B diritto: Sacchetto**	2 sem	
mfn0617	Economia e Gestione dell'Innovazione	secs-p/08	D	a scelta	6	**Pironti**	1 sem	
mfn0600	Elementi di Probabilità e Statistica	mat/06	C	affine e integrative	6	**A: Sirovich** **B: Sirovich, Giraudo**	1 sem	
mfn0598	Fisica	fis/01	C	affine e integrative	6	**A: Migliore, Gagliardi** **B: Pesando, Nocera**	2 sem	
mfn1353	Interazione Uomo Macchina	inf/01	D	a scelta	6	**Gena, Segnan**	1 sem	mutua su modulo di Interazione Uomo Macchina e Tecnologie Web
mfn0608	Interazione Uomo Macchina e Tecnologie Web	inf/01	B	caratterizzante	12	**Ardissono, Gena, Segnan**	1 sem	
mfn0590	Lingua Inglese I	L-Lin/12	E	conoscenza lingua straniera	3	**esercitatore Merrett (commissione: Radicioni, Cordero, Bono, Segnan, Gliozzi)**	1 e 2 sem	
mfn0610	Linguaggi e Paradigmi di Programmazione	inf/01	B	caratterizzante	9	**Bono**	1 sem	
mfn1354	Linguaggi e Paradigmi di Programmazione	inf/01	D	a scelta	6	**Bono**	1 sem	mutua su modulo di Linguaggi e Paradigmi di Programmazione da 9 CFU
mfn0603	Linguaggi Formali e Traduttori	inf/01	B	caratterizzante	9	**A: Sproston** **B: Pozzato** **Lab T1: Patti** **Lab T2: Sproston** **Lab T3: Patti** **Lab T4: Di Caro**	1 sem	
inf0003	Logica per l'Informatica	mat/01	D	a scelta	6	**Paolini**	1 e 2 sem	
Inf0291	Logica	Mat/01	A	Di base	6	**A: Andretta**	1 sem	

						B: Motto Ros **C: Viale**		
Inf0290	Matematica Discreta	mat/02	A	di base	6	**A: Chen, Terracini** **B: Mori** **C: Bertone, Terracini**	1 sem	
Inf0190	Metodi Formali dell'Informatica – parte A	Inf/01	D	A scelta	6	**de' Liguoro**	1 sem	mutua su modulo di Metodi Formali dell'Informatica da 9 cfu
mfn0633	Metodi Formali dell'Informatica	inf/01	B	caratterizzante	9	**de' Liguoro**	1 e 2 sem	
Inf0193	Metodologie e Tecnologie Didattiche per l'Informatica	Inf/01	D	A scelta	6	**Capecchi**	1 sem	
mfn0582	Programmazione I	inf/01	A	di base	9	**A: Cardone, Baroglio** **B: Roversi** **C: Beccuti** **Lab A1: Di Caro** **Lab A2: Mazzei** **Lab B1: Roversi** **Lab B2: Fiandrotti** **Lab C1: Basile** **Lab C2: Birke**	1 sem	
mfn0585	Programmazione II	inf/01	A	di base	9	**A: Bono** **B: Berardi** **C: Ciravegna** **Lab A1: Damiani** **Lab A2: Damiani** **Lab B1: Torta** **Lab B2: Berardi** **Lab C1: Ciravegna** **Lab C2: Magro**	2 sem	
mfn0605	Programmazione III	inf/01	B	caratterizzante	6	**Ardissono** **Lab 1 Ardissono** **Lab 2 Ardissono** **Lab 3 Esposito**	1 sem	
mfn0635	Reti di Elaboratori	inf/01	B	caratterizzante	12	**Garetto, Sereno**	1 e 2 sem	
mfn1362	Reti I	inf/01	B	caratterizzante	6	**Botta**	1 sem	
inf0002	Servizi Web	inf/01	D	a scelta	6	**Ardissono**	1 sem	mutua su modulo di Interazione Uomo Macchina e Tecnologie Web
mfn0636	Sicurezza	inf/01	B	caratterizzante	6	**Bergadano**	2 sem	
mfn0618	Sistemi Informativi	inf/01	D	a scelta	6	**Micalizio**	1	
mfn0607	Sistemi Intelligenti	inf/01	B	caratterizzante	6	**Baroglio**	2 sem	
mfn0601	Sistemi Operativi	inf/01	B	caratterizzante	12	**A: Gunetti** **B: Baroglio**	1 sem	

						Lab T1 C: De Pierro **Lab T1 Unix: Radicioni** **Lab T2 C: Bini** **Lab T2 Unix: Bini** **Lab T3 C: Bini** **Lab T3 Unix: Bini** **Lab T4 C: De Pierro** **Lab T4 Unix: Schifanella C**		
mfn0606	Sviluppo delle Applicazioni Software	inf/01	B	caratterizzante	9	**A: Baldoni** **B: Damiani** **lab 1: Capecchi** **lab 2: Picardi** **lab 3: Polato** **lab 4: Birke**	2 sem	
inf0004	Storia dell'Informatica	inf/01	D	a scelta	6	**Gunetti, Cardone**	1 e 2 sem	
mfn0634	Tecnologie Web	inf/01	B	caratterizzante	6	**A: Botta** **B: mutuato da A**	1 sem	
Inf0073	Stage		F	altre attività	9			
Inf0072	Prolungamento stage		D	a scelta	6			
Inf0074	Prova Finale		E	altre attività	3			

Codici per convalide di insegnamenti e competenze.

Gli studenti e le studentesse che per effetto di passaggio o trasferimento abbiano ottenuto la convalida di crediti liberi con nessuna corrispondenza specifica a insegnamenti del Corso di Laurea in Informatica, sono pregati di selezionare i codici contenitore MFN1522, MFN1409 e/o MFN1408 in base alla delibera della Commissione Passaggi e Trasferimenti, integrando eventualmente con altro insegnamento libero da selezionare dalla lista prevista per il percorso scelto, in caso non sia ancora stato raggiunto il minimo di 12 CFU. I codici Blended intensive programme sono selezionabili solo da chi partecipa a specifici progetti di mobilità.

mfn1409	Altre Attività	altre attività, a scelta lettera A	12
mfn1408	Altre Attività	altre attività, a scelta lettera A	6
mfn1522	Altre Attività	altre attività, a scelta lettera A	6
INT1530	Blended intensive programme		1
INT1531	Blended intensive programme		2
INT1524	Blended intensive programme		3
INT1525	Blended intensive programme		4
INT1526	Blended intensive programme		5
INT1527	Blended intensive programme		6

Informazioni aggiornate al 7 dicembre 2022

Syllabus degli insegnamenti

Insegnamento **MFN0597 - Algoritmi e Strutture Dati**

Insegnamento (inglese):	**Algorithms and Data Structures**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Giorgio AUDRITO (Titolare)** **Ugo DE' LIGUORO (Titolare)** **Andras HORVATH (Titolare)** **Diego MAGRO (Titolare)** **Roberto MICALIZIO (Titolare)** **Gian Luca POZZATO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si presuppone che lo studente sia a conoscenza delle basi della programmazione e dei linguaggi di programmazione Java e C; che sia in possesso delle nozioni elementari della matematica del discreto, del continuo, e della logica matematica.

Eventuali corsi propedeutici

Programmazione I & Laboratorio, Programmazione II & Laboratorio, Matematica Discreta, Analisi Matematica, Logica Matematica, Sistemi Operativi (in particolare l'apprendimento del linguaggio C).

2. Obiettivi formativi:

L’insegnamento ha lo scopo di introdurre i concetti e le tecniche fondamentali per l’analisi e la progettazione di algoritmi, che sono alla base dello sviluppo del software. Gli studenti acquisiranno conoscenze circa l’analisi di correttezza e complessità computazionale degli algoritmi, sulle strutture dati per la rappresentazione dell’informazione, sulle tecniche di problem-solving mediante lo sviluppo di algoritmi efficienti. L’insegnamento è supportato da un laboratorio che ne costituisce parte integrante, finalizzato alla realizzazione e sperimentazione degli algoritmi e delle strutture dati mediante un linguaggio imperativo ed uno object-oriented.

7. Programma:

Programma

- Problemi e algoritmi: risolubilità, correttezza, complessità.
- Analisi computazionale e complessità asintotica
- Algoritmi di ordinamento
    - Algoritmi elementari quadratici
    - Divide et impera: mergesort e quicksort
    - Risoluzione di relazioni di ricorrenza
    - Limiti inferiori per l’ordinamento
- Programmazione dinamica
    - Massima sottosequenza comune
- Strutture dati
    - Strutture concrete: array, liste, tabelle hash
    - Strutture astratte: pile, code, dizionari
    - Code di priorità, heapsort
    - Analisi ammortizzata, cenni
- Alberi

    - Definizione e visita
    - Alberi di ricerca
    - Alberi rosso-neri
- Grafi
    - Definizione e visita
    - Ordinamento topologico e componenti fortemente connesse
    - Algoritmi greedy: alberi di copertura minima
    - Cammini minimi: algoritmo di Dijkstra

Insegnamento **MFN0570 - Analisi Matematica**

Insegnamento (inglese):	**Mathematical Analysis**
CFU:	**9**
Settore:	**MAT/05 - ANALISI MATEMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Vivina Laura BARUTELLO (Titolare)** **Alberto BOSCAGGIN (Titolare)** **Gianluca GARELLO (Titolare)** **Joerg SEILER (Titolare)** **Stefano VITA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

L'insegnamento prevede la conoscenza dei contenuti di matematica di base forniti dalla scuola secondaria di secondo grado.

In particolare, a livello di conoscenze e comprensione in ingresso lo studente dovrà:

- conoscere i concetti di base sulla retta, sia dal punto di vista della geometria sintetica sia della geometria analitica, con particolare riferimento al concetto di pendenza;

- conoscere le funzioni quadratiche e le loro proprietà algebriche e grafiche;

- ricordare le proprietà delle potenze e dei logaritmi e conoscere i grafici delle funzioni potenza, esponenziali e logaritmiche;

- conoscere gli elementi essenziali di trigonometria (misure degli angoli in radianti, grafici delle funzioni circolari);

- conoscere i concetti di dominio, immagine, zeri, segno e monotonia per funzioni reali di una variabile reale.

Inoltre, come applicazione di conoscenza e comprensione, lo studente dovrà saper:

- determinare l'equazione della retta passante per un punto ed avente pendenza assegnata e l'equazione della retta passante per due punti;

- determinare l'equazione di una retta a partire dal suo grafico (calcolo di pendenza e intercetta);

- tracciare il grafico di una funzione lineare e determinare per via grafica dominio, immagine, zeri e segno, monotonia;

- tracciare il grafico di una funzione quadratica e determinare per via grafica dominio, immagine, zeri e segno, monotonia;

- riconoscere come varia la retta tangente al grafico di una funzione quadratica in un suo punto, anche in relazione alla concavità della funzione;

- risolvere equazioni e disequazioni di primo e secondo grado, anche per via grafica;

- tracciare il grafico di funzioni potenza xa, con a intero positivo o negativo, e determinarne per via grafica dominio, immagine, simmetrie, zeri e segno, monotonia, massimi e minimi;

- tracciare il grafico di funzioni del tipo a^x o log_a x, con a positivo, e determinarne per via grafica dominio, immagine, zeri e segno, monotonia;

- risolvere equazioni e disequazioni del tipo a^x=b, a^x>b, a^x <b, log_a x=b, log_a x>b e log_a x<b;

- trasformare la misura di un angolo da gradi a radianti e viceversa;

- tracciare il grafico delle funzioni circolari;

- risolvere equazioni del tipo sin x =b e cos x=b;

- determinare dominio, immagine, zeri e segno, monotonia di una funzione a partire dal suo grafico.

Eventuali corsi propedeutici

I prerequisiti richiesti potranno essere recuperati attraverso il Corso di Riallineamento di Matematica presente sulla Piattaforma Orient@mente.

2. Obiettivi formativi:

L'insegnamento ha lo scopo di presentare le nozioni di base su funzioni, grafici e loro trasformazioni, di introdurre i concetti di derivata e di integrale definito e di illustrare l'utilizzo di tecniche di tipo analitico nello studio di fenomeni discreti. Si tratta di argomenti indispensabili per la formazione dei laureati in scienze dell'informazione (classe L-31).

L'insegnamento concorre agli obiettivi formativi dell'area Matematico-Fisica del corso di Laurea in Informatica, fornendo conoscenze relative ai concetti ed agli strumenti matematici e metodologici fondamentali necessari per descrivere, schematizzare e interpretare i principali aspetti della realtà che ci circonda. In particolare, l'insegnamento si inoltre propone di accrescere le capacità di comprensione degli studenti e di consentire loro di acquisire un modo rigoroso ed analitico di ragionare e affrontare nuovi problemi.

La significativa presenza di teoremi, molti dei quali con dimostrazione, ha l'obiettivo di rafforzare nello studente le attitudini logico-deduttive apprese nel corso di Matematica discreta e Logica.

7. Programma:

1. Funzioni, grafici e modelli

1.1. Funzioni elementari e loro grafici

1.2. Trasformazioni geometriche di grafici

1.3. Grafici di funzioni composte

2. Il concetto di limite

2.1. Il concetto di limite nel caso continuo (limite di funzioni)

2.2. Il concetto di limite nel caso discreto (limite di successioni)

2.3. Principali risultati teorici sui limiti

2.4. Successioni definite per ricorrenza

2.5. Crescite e confronti di crescite: i simboli di Landau

3. Calcolo differenziale

3.1. Derivata di una funzione in un punto

3.2. Funzione derivata e funzioni primitive; relazioni tra una funzione e la sua derivata o le sue primitive

3.3. Derivata e monotonia; derivata e convessità

3.4. Approssimazione locale di funzioni mediante polinomi

4. Risoluzione approssimata di equazioni

4.1. Il Teorema di esistenza degli zeri ed il metodo di bisezione

4.2. Il metodo di Newton

5. Calcolo integrale

5.1. Integrale definito di una funzione su un intervallo

5.2. Teorema Fondamentale del Calcolo Integrale

5.3. Teorema di Torricelli-Barrow

5.4. Integrali impropri

6. Serie numeriche

6.1. La serie geometrica

6.2. Le serie armoniche generalizzate

6.3. Definizioni e risultati teorici sulle serie

6.4. Confronto tra serie ed integrali impropri

Insegnamento **MFN0586 - Architettura degli Elaboratori**

Insegnamento (inglese):	**Computer architecture**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Idilio DRAGO (Titolare)** **Rossano GAETA (Esercitatore)** **Rossano GAETA (Titolare)** **Michele GARETTO (Titolare)** **Maurizio LUCENTEFORTE (Titolare)** **Daniele Paolo RADICIONI (Titolare)** **Claudio SCHIFANELLA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza di base di programmazione imperativa, capacità di progettare semplici algoritmi

Eventuali corsi propedeutici

È richiesta la conoscenza delle competenze fornite dai corsi di:

- Programmazione I e Laboratorio (I semestre)
- Matematica Discreta e Logica (I semestre)

2. Obiettivi formativi:

L’insegnamento si propone di fornire allo studente:

- la conoscenza delle tecniche di codifica dell’informazione all’interno dei calcolatori;
- la conoscenza dell’organizzazione hardware degli elaboratori, attraverso la nozione di gerarchie di macchine virtuali;
- la comprensione delle funzioni svolte dall’hardware e utilizzate dai sistemi operativi;
- le nozioni di base per la comprensione del processo di traduzione da linguaggi ad alto livello a linguaggio macchina.

L'insegnamento è supportato da un laboratorio, finalizzato alla programmazione utilizzando un linguaggio assemblativo composto da un insieme di istruzioni appartenente allo standard RISC-V.

7. Programma:

Teoria (6 CFU - 48 ore)

- Introduzione. Calcolatori: astrazioni e tecnologia
- Le istruzioni e la rappresentazione dell'informazione
    - RISC-V ISA
    - Bit, byte, parola, operandi dell'hardware e gli indirizzi
    - Rappresentazione numerica, numeri con e senza segno, complemento a due
    - Rappresentazione delle istruzioni nel calcolatore
    - La rappresentazione di informazione non numerica (e.g., le stringhe)
    - Traduzione e avvio di un programma (assembler, linker, loader)
    - L'aritmetica dei calcolatori e i numeri in virgola mobile
- Sistemi digitali

    - Porte logiche, tabella di verità e espressioni Booleane
    - Circuiti combinatori
    - Unità aritmetica e logica
    - Clocks
    - Memoria: Flip-Flops, Latches, e registri
    - Memoria: SRAMs and DRAMs
    - Automa a stati finiti
- Il processore RISC-V
    - Introduzione di base del RISC-V
    - Metodologia di temporizzazione
    - Realizzazione di un'unità di elaborazione
    - Uno schema semplice di implementazione
- Bus e I/O
    - I tipi di bus e l'arbitraggio
    - I/O programmato con busy waiting (pooling)
    - I/O controllato dall'interrupt
    - Direct Memory Access
    - Il meccanismo di interruzione del RISC-V
- Gerarchia delle memorie
    - Organizzazione delle memorie, località temporale e spaziale
    - Tecnologie delle memorie
    - Principi base e prestazioni delle memorie cache

Laboratorio (3 CFU - 30 ore)

- Rappresentazione numerica
- RISC-V ISA e linguaggio assemblativo
- Operandi allocati in memoria, indirizzi e registri
- Operazioni logiche
- Istruzioni condizionali - if-then-else e cicli
- Supporto hardware alle procedure - registri, stack e heap
- Compilazione e traduzione dei programmi
- Circuiti combinatori e sequenziali

Insegnamento **MFN0602 - Basi di Dati**

Insegnamento (inglese):	**Databases**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Luca ANSELMA (Titolare)** **Noemi MAURO (Titolare)** **Ruggero Gaetano PENSA (Titolare)** **Fabiana VERNERO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Lo studente deve avere familiarità con i concetti fondamentali della teoria degli insiemi e della logica. Deve aver acquisito capacità di progettare algoritmi fondamentali, programmare in linguaggi ad alto livello e possedere nozioni di base di sistemi operativi.

Eventuali corsi propedeutici

Insegnamenti di Logica, Matematica Discreta e Programmazione 1 e 2, Sistemi Operativi.

2. Obiettivi formativi:

L'insegnamento concorre al raggiungimento degli obiettivi formativi del Corso di Laurea in Informatica fornendo un'introduzione alle basi di dati e ai sistemi di gestione delle medesime (DBMS). Si propone perciò di fornire le competenze teoriche e pratiche necessarie alla progettazione e gestione delle basi di dati, e la padronanza degli strumenti per analizzare, progettare, sviluppare e gestire le basi di dati.

7. Programma:

Fondamenti teorici delle basi di dati relazionali (5 CFU, 40 ore):

- Introduzione ai sistemi informativi e alle basi di dati
- Il modello relazionale
- L'algebra relazionale, con progetto di espressioni in algebra
- Introduzione al calcolo relazionale
- Introduzione alle transazioni
- Ridondanze e anomalie, dipendenze funzionali e teoria della normalizzazione
- Introduzione all’architettura dei sistemi di gestione delle basi di dati: memorizzazione dei dati, indipendenza dei dati, affidabilità, problemi di concorrenza

Laboratorio di Progettazione ed SQL (4 CFU, 40 ore):

- Laboratorio di Progettazione
    1. Metodologie di progetto
    2. Il modello Entity Relationship
    3. La progettazione concettuale
    4. La progettazione logica
    5. Cenni di progettazione fisica
- Laboratorio di SQL con Oracle/PostgreSQL
    1. Il linguaggio SQL e gli standard
    2. Definizione dei dati in SQL (DDL)
    3. Modifica dei dati in SQL (DML)

4. Interrogazioni in SQL (DML/DQL) e progettazione delle espressioni in SQL

Insegnamento **INF0090 - Calcolabilità e Complessità**

Insegnamento (inglese):	**Computability and Complexity**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante** **D - libera**
Docenti:	**Stefano BERARDI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Sono richieste buone conoscenze di logica, programmazione e di algoritmi. E' utile ma non indispensabile che lo studente possegga le nozioni di linguaggio formale, grammatica e di automa.

Eventuali corsi propedeutici

Matematica Discreta e Logica, Programmazione 1 e 2, Algoritmi e Strutture Dati, Linguaggi Formali e Traduttori.

2. Obiettivi formativi:

Che cos'e' un algoritmo? Quali problemi si possono risolvere con un algoritmo? E in quali casi un algoritmo richiede risorse inaccessibili nella pratica? Il corso affronta questi problemi, trattando anzitutto la teoria della computabilita' sia dal punto di vista matematico - macchine di Turing, funzioni ricorsive - che da prospettive legate ai linguaggi di programmazione. Si discutono poi i vari possibili criteri di misura delle risorse disponibili (tempo, memoria) e le classi di complessita'. Finiamo presentando il problema P = NP .

7. Programma:

Teoria della computabilita'

- Le Macchine di Turing
- Problemi non risolubili
- Funzioni ricorsive
- Calcolabilita' e Linguaggi di Programmazione

Teoria della complessita'

- Misure e classi di Complessita'
- Classi di Complessita' Temporale
- Classi di Complessita' Spaziale
- Le classi P ed NP
- Problemi NP completi

Definizione di PSPACE e NPSPACE. Teorema di Savich. Inclusioni P in NP in PSPACE in EXPTIME. Classi L e NL e inclusioni note.

Insegnamento	**MFN0588 - Calcolo Matriciale e Ricerca Operativa**

Insegnamento (inglese):	**Matrix Calculus and Operational Research**
CFU:	**6**
Settore:	**MAT/09 - RICERCA OPERATIVA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Roberto ARINGHIERI (Titolare)** **Andrea Cesare GROSSO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Nozioni di base di algebra e insiemistica comuni nei programmi di matematica della scuola superiore.

Eventuali corsi propedeutici

Nessuno.

2. Obiettivi formativi:

Il corso si propone di fornire agli studenti nozioni generali di calcolo matriciale, algebra e geometria, e nozioni più specifiche di ricerca operativa.

Il calcolo matriciale è uno strumento fondamentale per il calcolo scientifico. La ricerca operativa studia modelli e metodi, basati sulle tecniche introdotte, per l'utilizzo ottimale di risorse scarse (in ambiti produttivi, finanziari, ecc.).

7. Programma:

1. Vettori e matrici. Operazioni fondamentali. Insiemi convessi, poliedri.
2. Soluzione di un sistema di equazioni lineari.
3. Combinazioni lineari, indipendenza lineare.
4. Programmazione lineare.
    - Modellazione.
    - Struttura della regione ammissibile. Metodo grafico di soluzione.
    - Soluzioni di base. Algoritmo del simplesso

Insegnamento	**MFN0604 - Economia e Gestione dell'Impresa e Diritto**

Insegnamento (inglese):	**Management, business administration and computer law**
CFU:	**9**
Settore:	**IUS/02 - DIRITTO PRIVATO COMPARATO** **SECS-P/08 - ECONOMIA E GESTIONE DELLE IMPRESE**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**C - affine e integrativa**
Docenti:	**Lea IAIA (Titolare)** **Fabio MONTALCINI (Professore a Contratto)** **Marco PIRONTI (Titolare)** **Camillo SACCHETTO (Professore a Contratto)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Il corso non presuppone conoscenze iniziali specifiche

Eventuali corsi propedeutici

NESSUNO

2. Obiettivi formativi:

Modulo di Economia

l'obiettivo del corso è analizzare le caratteristiche basi di una azienda: dal modello di business teorico alla creazione dei processi produttivi, alla comunicazione e vendita dei prodotti.

Verranno studiate le relazioni delle aziende all'interno della catena del valore (Clieti /fornitori) e i loro settori di riferimento. I framework teorici saranno poi applicati per l'analisi di aziende, settori e mercati innovativi al fine di valutare come vengono colte le opportunità che l'innovazione dà alle aziende e ipotizzare sviluppi futuri di modelli. durante il corso gli studenti si confronteranno con casi aziendali reali e con imprenditori. gli studeneti del corso potranno inoltre partecipare al progetto silicon valley study tour e passare una settimana in Silicon Valley dove conoscere le aziende più innovative dell'IT.

Modulo di Diritto

Agli studenti saranno forniti le nozioni, i metodi, i modelli cognitivi e gli strumenti tecnici necessari alla comprensione dei concetti fondamentali del Diritto dell'Informatica, settore della scienza forense che realizza maggiormente il dialogo teorico e la sintesi operativa tra il diritto (in particolar modo nei suoi profili pratici e processuali) e l'informatica. Durante il Corso si analizzeranno i principali temi della materia anche alla luce dei piu' significativi casi nazionali e internazionali.

7. Programma:

Modulo di Economia

business model struttura dell'organizzazione struttura dei processi struttura delle funzioni strategie analisi di settore analisi di mercato analisi unita' organizzativa e processi aziendali

Modulo di Diritto

Responsabilità Internet Service Provider; Computer Crimes e Prove Digitali; Documento informatico, Firme elettroniche e Posta Elettronica Certificata (PEC); Proprietà intellettuale in ambiente web; Privacy: Principi generali e suoi ambiti peculiari; E-commerce e web contracts.

Insegnamento

MFN0617 - Economia e Gestione dell'Innovazione

Insegnamento (inglese):	**Managing Innovation**
CFU:	**6**
Settore:	**SECS-P/08 - ECONOMIA E GESTIONE DELLE IMPRESE**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Marco PIRONTI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza dei concetti base di Economia e Gestione delle Imprese: analisi di settore analisi di mercato/consumatore modello di business analisi delle organizzazioni strategie delle aziende

Eventuali corsi propedeutici

Corso di Economia e Gestione delle Imprese

2. Obiettivi formativi:

Questo corso ha l'obiettivo di fornire gli skill, tools e approcci per la creazione di un' innovazione sostenibile e di successo nell'era digitale. il corso è multidisciplinare e costruito sulle ultime ricerche legate ai framework di gestione dell'innovazione, sulle metodologie lean per la creazione dei prototipi, sull'additive manufacturing e sull'approccio del design thinking. Insieme, esploreremo come identificare le opportunità dell'economia digitale e creare un'innovazione che va ben oltre il semplice ricopiare o reinventare un prodotto o servizio risolvendo problemi reali della nostra società quali creazione di lavoro, diminuzione di side effect eliminazione di colli di bottiglia all'interno di prodotti, modelli e settori. identificheremo inoltre framework informatici per facilitare l'identificazione,lo sviluppo e la creazione di innovazione in modo economico e sostenibile. l'obiettivo del corso non sarà solo acquistare gli strumenti teorici per creare un innovazione ma anche riuscire ad applicarli per arrivare a creare un'innovazione di prodotto o servizio all'interno di un mercato di riferimento

7. Programma:

- l'analisi strategica - le nuove strategie per l'innovazione - l'analisi dei modelli di business innovativi -il consumatore - l'individuazione dell'opportunità - la value proposition - creazione del prototipo - Test Lean - sperimentazione - design/comunicazione e visualizzazione

Insegnamento **MFN0600 - Elementi di Probabilità e Statistica**

Insegnamento (inglese):	**Foundations of Probability and Statistics**
CFU:	**6**
Settore:	**MAT/06 - PROBABILITA\' E STATISTICA MATEMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**C - affine e integrativa**
Docenti:	**Maria Teresa GIRAUDO (Titolare)** **Roberta SIROVICH (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Calcolo di base.

2. Obiettivi formativi:

L’insegnamento si inserisce nell’area tematica della matematica, in particolare della probabilità e statistica. Si propone il duplice obiettivo di fornire allo studente sia la conoscenza e la capacità di comprensione dei fenomeni di natura aleatoria sia gli strumenti metodologici e analitici correlati, che siano di supporto per i corsi successivi ma anche di valore intrinseco. L’introduzione ai concetti di rischio e di probabilità fornisce gli strumenti analitici e modellistici per la trattazione di eventi casuali. L’introduzione alla statistica fornisce gli strumenti metodologici per trattare con le quantità aleatorie rilevabili.

7. Programma:

Il programma di lavoro è articolato in due parti, la probabilità e la statistica.
Probabilità. Spazio campionario e probabilità, insiemi, modelli probabilistici, probabilità condizionata, teorema delle probabilità totali, formula di Bayes, indipendenza. Variabili aleatorie discrete, funzione di densità discreta, funzioni di variabili aleatorie, attesa, varianza. Densità discreta congiunta di variabili multidimensionali, condizionamento e indipendenza. Variabili aleatorie continue, funzione di densità, funzione di distribuzione cumulata. Densità di probabilità variabili aleatorie multidimensionali. Covarianza e correlazione. Legge dei grandi numeri e teorema del limite centrale.
Statistica. Introduzione a R. Dati univariati, tipi di dati, indici riassuntivi di posizione, di variabilità e di forma. Rappresentazioni grafiche, istogrammi e box plot. Dati invariati, scatterplot. Confronti qualitativi, dati appaiati. Dati invariati categoriali e tabelle. Intervalli di confidenza (proporzioni, medie, varianze, differenze di medie). Test di ipotesi (proporzioni, medie).

Insegnamento **MFN0598 - Fisica**

Insegnamento (inglese):	**Physics**
CFU:	**6**
Settore:	**FIS/01 - FISICA SPERIMENTALE**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**C - affine e integrativa**
Docenti:	**Martino GAGLIARDI (Titolare)** **Ernesto MIGLIORE (Titolare)** **Emanuele Roberto NOCERA (Titolare)** **Igor PESANDO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Competenze di base di trigonometria, di calcolo vettoriale e di analisi matematica.

Eventuali corsi propedeutici

Insegnamenti di Matematica del I anno.
Nella prima settimana di lezione si svolgerà un ciclo di lezioni (8h) di richiamo sulla *Meccanica del punto materiale* (argomenti trattati: richiami di cinematica; leggi di Newton; lavoro ed energia; forze conservative).

2. Obiettivi formativi:

L'insegnamento si propone di fornire le conoscenze necessarie per approfondire la struttura fisica dei calcolatori e dei circuiti che li compongono. Nello specifico:

1. introdurre alla conoscenza delle basi indispensabili di meccanica, delle principali proprietà del campo elettrico e del campo magnetico, con cenni al comportamento della materia soggetta a tali campi;
2. introdurre alla conoscenza del comportamento degli elementi di un circuito in corrente continua;
3. introdurre alla conoscenza dei principi fisici alla base del funzionamento delle porte logiche.

7. Programma:

Il campo elettrostatico. Legge di Gauss. Conservatività del campo elettrostatico. Superfici equipotenziali. Conduttori e dielettrici. Capacità elettrica di un conduttore. Condensatori. Densità di energia del campo elettrico. Correnti elettriche. Leggi di Ohm e di Kirchhoff. Circuiti RC.

Fisica dei dispositivi elettronici. Conduttori, isolanti e semiconduttori. Giunzione pn e diodo. Dispositivi MOS e porte logiche (NOT, NOR, NAND).

Il campo magnetico indipendente dal tempo. Magneti. Moto di una carica in campo magnetico; esempi ed applicazioni. Filo percorso da corrente in campo magnetico. Campo magnetico generato da un filo percorso da corrente. Legge di Ampere. Legge di Gauss per il campo magnetico.

Campi elettrici e magnetici variabili nel tempo. Induzione elettromagnetica. Legge di Faraday-Henry. Correnti alternate. Legge di Ampere-Maxwell. Autoinduzione. Induttanza del

solenoide ideale. Densità di energia del campo magnetico. Circuiti RL.

Insegnamento	**MFN1353 - Interazione Uomo Macchina**

Insegnamento (inglese):	**Human-computer Interaction**
CFU:	**6**
Settore:	
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Cristina GENA (Titolare)** **Marino SEGNAN (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si richiede una buona conoscenza delle basi di dati (fornita dal corso di Basi di Dati), dell'analisi e della progettazione object-oriented (fornita dai corsi di Programmazione II e di Algoritmi e strutture dati) e dei fondamenti della programmazione distribuita (fornita dal corso di Programmazione III). Lo studente deve inoltre avere la capacità di scrivere, compilare e verificare la correttezza di programmi in Java.

Eventuali corsi propedeutici

Basi di dati e sperimentazioni, Algoritmi e sperimentazioni, Programmazione II and III.

2. Obiettivi formativi:

Nella progettazione e sviluppo di un’applicazione software si devono tenere in conto due aspetti fondamentali: (i) l’interazione con l’utente che, indipendentemente dall’efficacia delle funzionalità offerte da un’applicazione, influenza sensibilmente il suo successo in termini di utilizzo. L’applicazione deve essere usabile ed accessibile per permettere un uso agevole del software e prevenirne l’abbandono da parte degli utenti. (ii) l’implementazione dell’applicativo, che richiede di guardare “dietro all’interfaccia utente” per andare a fondo su aspetti architetturali e tecnologici che possono influenzare non solo le prestazioni dell’applicazione, la sua scalabilità e robustezza, ma anche le tipologie di servizio che possono effettivamente essere offerte.

Partendo da queste considerazioni, il corso si pone un duplice obiettivo: da una parte, fornire la conoscenza di base necessaria per il disegno corretto di interazioni uomo-macchina, che sta alla base della progettazione di applicazioni di ogni genere (web e non, mobili, etc.). Dall’altra, concentrandosi sulle applicazioni mobili, che sono oggetto specifico del corso, fornire la conoscenza di base necessaria per la progettazione e lo sviluppo di applicazioni interattive, accessibili da terminali mobili (come per esempio gli smart phone), e caratterizzate da una logica applicativa mediamente complessa.

Più precisamente, per quanto riguarda l'interazione uomo-macchina, verranno acquisite competenze sia teoriche sia pratiche nel disegno corretto di interazioni, con specifico riferimento alla progettazione user-centered. Per quanto riguarda gli aspetti architetturali e tecnologici, il corso tratterà dal punto di vista sia teorico che pratico la programmazione lato client per device mobili su piattaforma Android e lo sviluppo di interfacce mobili. Per permettere agli studenti di sperimentare le nozioni apprese durante le ore di teoria in aula il corso include una sostanziale parte di laboratorio. I temi introdotti durante il laboratorio corredano e integrano le conoscenze derivanti dalla parte teorica (knowledge and understanding) e permettono agli studenti di familiarizzare con le metodologie e tecnologie introdotte, anche investigando soluzioni alternative (applying knowledge and understanding).

Inoltre durante le ore di laboratorio è previsto lo sviluppo di un’applicazione realistica con interfaccia mobile ed una introduzione al linguaggio Python. La preparazione e la discussione del progetto di laboratorio sono inoltre volte a stimolare le capacità di organizzare il lavoro in piccoli

gruppi (max 4 studenti), e poi di illustrare verbalmente le soluzioni adottate (communication skills).

7. Programma:

- Parte Ia - Human-computer interaction (HCI)
- I concetti e ai principi dell'Interaction Design e della User Experience (UX)
- Il processo di UX e il rapid prototyping
- Le linee guida di usabilità per il web (desktop e mobile)
- Principi di architettura dell'informazione
- Le linee guida di accessibilità internazionali e nazionali: le WCAG, le linee guida AGID
- Il design centrato sull'utente e le metodologie di valutazione: la fase di valutazione preliminare, la fase di valutazione formativa e la fase finale.
- La valutazione dell'accessibilità, sia oggettiva che soggettiva

- Parte Ib - Programmazione di device mobili.
    - Il sistema operativo Android e le sue peculiarità
    - Gli elementi costitutivi di una applicazione utente ed il suo ciclio di vita.
    - Le pratiche consolidate e l'ambiente di sviluppo di una applicazione Android.
    - Utilizzo del framework Flutter per la scrittura dei tipi di applicazioni più utilizzati
    - Il linguaggio avanzato Python: principali differenze rispetto al linguaggio Java e caratteristiche più interessanti

Insegnamento	**MFN0608 - Interazione Uomo Macchina e Tecnologie Web**

Insegnamento (inglese):	**Human-computer Interaction and Web Technologies**
CFU:	**12**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Liliana ARDISSONO (Titolare)** **Cristina GENA (Titolare)** **Marino SEGNAN (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si richiede una buona conoscenza delle basi di dati (fornita dall'insegnamento Basi di Dati), dei sistemi operativi (fornita dall'insegnamento Sistemi Operativi), dell'analisi e della progettazione object-oriented (fornita dagli insegnamenti Programmazione II, e Algoritmi e strutture dati) e dei fondamenti della programmazione distribuita (fornita dall'insegnamento Programmazione III). Gli studenti e le studentesse devono inoltre avere la capacità di scrivere, compilare e verificare la correttezza di programmi in Java.

Eventuali corsi propedeutici

Basi di Dati, Sistemi Operativi, Programmazione II, Algoritmi e strutture dati, Programmazione III

2. Obiettivi formativi:

Nella progettazione e sviluppo di un'applicazione software si devono tenere in conto due aspetti: (i) l'interazione con l'utente influenza sensibilmente il suo successo. L'applicazione deve essere usabile ed accessibile per permettere un uso agevole del software ed evitare che gli utenti la abbandonino. (ii) l'implementazione dell'applicativo richiede di guardare "dietro all'interfaccia utente" per andare a fondo su aspetti architetturali e tecnologici che possono influenzare non solo le prestazioni, la scalabilità e la robustezza, ma anche le tipologie di servizio effettivamente erogabili.

Date queste premesse, l'insegnamento si pone un duplice obiettivo: da una parte, fornire la conoscenza di base necessaria per il disegno corretto di interazioni uomo-macchina, che sta alla base della progettazione di applicazioni di ogni genere (web e non, mobili, etc.). Dall'altra, fornire la conoscenza di base necessaria per la progettazione e lo sviluppo di applicazioni Web interattive, accessibili da terminali desktop e mobili (come gli smart phone), e caratterizzate da una logica applicativa mediamente complessa.

Per quanto riguarda l'interazione uomo-macchina, verranno affrontate le tematiche fondamentali per una progettazione web (desktop/mobile) usabile, accessibile e centrata sull'utente e relative al processo di user experience, così come i principi e linee guida e la loro applicazione in scenari reali. Questi ultimi riguarderanno lo sviluppo di interfacce per il desktop e mobile web, per passare poi ad esaminare le principali metodologie di valutazione predittiva, formativa, e sommativa. Verranno di conseguenza affrontati gli approcci orientati al miglioramento della user experience in un'ottica di design centrato sull'utente. Per quanto riguarda gli aspetti architetturali e tecnologici, nell'insegnamento verranno acquisite competenze teoriche e pratiche su: (i) lo sviluppo di pagine web statiche e dinamiche responsive; (ii) la programmazione lato server; (iii) la programmazione lato client per device mobili su piattaforma Android e lo sviluppo di interfacce mobili per applicazioni web. Inoltre, l'insegnamento fornirà conoscenza relativa alla rappresentazione di dati con XML, data l'importanza per l'interoperabilità tra applicazioni.

In questo insegnamento, le capacità di applicare le conoscenze acquisite vengono sviluppate nell'ambito di esercitazioni sperimentali che permettono di applicare le nozioni apprese durante le ore di teoria, organizzando il lavoro in piccoli gruppi. I temi introdotti durante le esercitazioni corredano e integrano le conoscenze derivanti dalla parte teorica (knowledge and understanding) e permettono agli studenti e alle studentesse di familiarizzare con le metodologie e tecnologie introdotte, anche investigando soluzioni alternative (applying knowledge and understanding). Inoltre è previsto lo sviluppo di un'applicazione SW realistica con interfaccia sia web che mobile (denotata come PROGETTO SW), che va oltre la complessità del tipico esercizio svolto nelle esercitazioni. La progettazione e lo sviluppo del progetto SW permettono di allenare l'autonomia di giudizio nella scelta delle implentazioni da realizzare, analizzando in forma critica le informazioni acquisite nella parte teorica dell'insegnamento. La preparazione del progetto SW è anche volta a stimolare la capacità comunicativa degli studenti e delle studentesse che sono invitati/e a illustrare verbalmente le soluzioni adottate (communication skills).

7. Programma:

PARTE 1A: HUMAN-COMPUTER INTERACTION (HCI)

- I concetti e ai principi dell'Interaction Design e della User Experience (UX)
- Il processo di UX e il rapid prototyping
- Le linee guida di usabilità per il web (desktop e mobile)
- Principi di architettura dell'informazione
- Le linee guida di accessibilità internazionali e nazionali: le WCAG, le linee guida AGID
- Il design centrato sull'utente e le metodologie di valutazione: la fase di valutazione preliminare, la fase di valutazione formativa e la fase finale.
- La valutazione dell'accessibilità, sia oggettiva che soggettiva

PARTE 1B - PROGRAMMAZIONE DI DEVICE MOBILI

- Il sistema operativo Android e le sue peculiarità. Gli elementi costitutivi di una applicazione utente ed il suo ciclio di vita. Le pratiche consolidate e l'ambiente di sviluppo di una applicazione Android. Utilizzo del framework Flutter per la scrittura dei tipi di applicazioni più utilizzati
- Il linguaggio avanzato Python: principali differenze rispetto al linguaggio Java e caratteristiche più interessanti

PARTE 2 - APPLICAZIONI WEB E TECNOLOGIE DI SUPPORTO

- Architetture delle applicazioni Web: Web browser e Web server
- Il Pattern Model View Controller per le applicazioni Web - Progettazione e sviluppo di applicazioni Web a 3 livelli basate su MVC:
    - Il primo livello (client dell'applicazione): HTML5, CSS, scripting lato client (JavaScript e AJAX). Raccolta dati con HTML form
    - Il secondo livello (logica applicativa): Servlet Java
    - Il terzo livello (livello dei dati): accesso a database relazionali con Java Database Connectivity (JDBC), rappresentazione e gestione di informazioni con XML (XML Schema, XPath, XML Parsers)
- Framework le applicazioni web MVC (Vue.js)
- Introduzione al linguaggio PHP per lo sviluppo di pagine web dinamiche con accesso a DB relazionale

Insegnamento	**MFN0590 - Lingua Inglese I**

Insegnamento (inglese):	**English I**
CFU:	**3**
Settore:	**L-LIN/12 - LINGUA E TRADUZIONE - LINGUA INGLESE**
Periodo didattico:	**1** **2**
Tipologia di Attività Formativa:	**E - prova finale e lingua straniera**
Docenti:	**Viviana BONO (Titolare)** **Francesca CORDERO (Titolare)** **Valentina GLIOZZI (Titolare)** **Catherine Mary MERRETT (Professore a Contratto)** **Daniele Paolo RADICIONI (Titolare)** **Marino SEGNAN (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Nessuna.

2. Obiettivi formativi:

Esercitazioni di base di Inglese orientate alla comprensione dei testi a alla grammatica di base.

7. Programma:

- Grammatica di base
- Lessico
- Pratica

Informazioni dettagliate su https://informatica.i-learn.unito.i....

Insegnamento

MFN1354 - Linguaggi e Paradigmi di Programmazione

Insegnamento (inglese):	**Programming languages and paradigms**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Viviana BONO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza delle basi della matematica discreta e della programmazione.

Eventuali corsi propedeutici

Matematica discreta e logica; Programmazione 1 e 2.

2. Obiettivi formativi:

L'insegnamento si propone di fornire una introduzione al paradigma di programmazione funzionale, ai sistemi di tipi per i linguaggi funzionali con i relativi algoritmi di inferenza e alla dimostrazione di correttezza di programmi funzionali. Nella parte avanzata del corso si introdurranno tecniche di programmazione e dimostrazione per strutture dati co-induttive - in particolare gli stream - e si illustrerà l'efficacia delle tecniche di programmazione funzionale apprese quando applicate a Java, un moderno linguaggio multi-paradigma che a partire dalla versione 8 ha incorporato costrutti tipici della programmazione funzionale, in particolare le lambda espressioni.

7. Programma:

Nel seguente elenco di argomenti non viene fatta distinzione tra argomenti svolti in aula ed argomenti svolti in laboratorio, che sono peraltro strettamente connessi.

- Breve storia dei linguaggi di programmazione, con particolare riferimento a quelli funzionali;
- Calcolo come riscrittura: le basi dell'esecuzione dei programmi funzionali;
- Espressioni e loro tipi. Tipi di base;
- Progettazione di programmi funzionali. Tecniche di ricorsione;
- Liste e funzioni del prim'ordine su liste;
- Dimostrazioni di correttezza di programmi funzionali usando l'induzione;
- L'idea di astrazione funzionale. Funzioni di ordine superiore e pattern di trasformazione;
- Alberi e tipi algebrici generali.

Per la versione da 9 CFU, i seguenti argomenti addizionali:

- Cenni a monadi e programmazione imperativa;
- Strutture di dati infinite (stream) e dimostrazioni di correttezza usando la co-induzione;
- Programmazione funzionale in Java 8.

Insegnamento

MFN0610 - Linguaggi e Paradigmi di Programmazione

Insegnamento (inglese):	**Programming languages and paradigms**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Viviana BONO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza delle basi della matematica discreta e della programmazione.

Eventuali corsi propedeutici

Matematica discreta e logica; Programmazione 1 e 2.

2. Obiettivi formativi:

L'insegnamento si propone di fornire una introduzione al paradigma di programmazione funzionale, ai sistemi di tipi per i linguaggi funzionali con i relativi algoritmi di inferenza e alla dimostrazione di correttezza di programmi funzionali. Nella parte avanzata del corso si introdurranno tecniche di programmazione e dimostrazione per strutture dati co-induttive - in particolare gli stream - e si illustrerà l'efficacia delle tecniche di programmazione funzionale apprese quando applicate a Java, un moderno linguaggio multi-paradigma che a partire dalla versione 8 ha incorporato costrutti tipici della programmazione funzionale, in particolare le lambda espressioni.

7. Programma:

Nel seguente elenco di argomenti non viene fatta distinzione tra argomenti svolti in aula ed argomenti svolti in laboratorio, che sono peraltro strettamente connessi.

- Breve storia dei linguaggi di programmazione, con particolare

riferimento a quelli funzionali;

- Calcolo come riscrittura: le basi dell'esecuzione dei programmi

funzionali;

- Espressioni e loro tipi. Tipi di base;
- Progettazione di programmi funzionali. Tecniche di ricorsione;
- Liste e funzioni del prim'ordine su liste;
- Dimostrazioni di correttezza di programmi funzionali usando

l'induzione;

- L'idea di astrazione funzionale. Funzioni di ordine superiore e

pattern di trasformazione;

- Alberi e tipi algebrici generali.

Per la versione da 9 CFU, i seguenti argomenti addizionali:

- Cenni a monadi e programmazione imperativa;
- Strutture di dati infinite (stream) e dimostrazioni di correttezza

usando la co-induzione;

- Programmazione funzionale in Java 8.

Insegnamento **MFN0603 - Linguaggi Formali e Traduttori**

Insegnamento (inglese):	**Formal Languages and Compilers**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Luigi DI CARO (Titolare)** **Viviana PATTI (Titolare)** **Gian Luca POZZATO (Titolare)** **Jeremy James SPROSTON (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Familiarità con i concetti fondamentali della logica, della teoria degli insiemi e della progettazione di algoritmi iterativi e ricorsivi. Capacità di programmare in linguaggi ad alto livello.

Eventuali corsi propedeutici

Programmazione I; Programmazione II; Architettura degli elaboratori; Matematica Discreta e Logica.

2. Obiettivi formativi:

L'insegnamento ha l'obiettivo di fornire le nozioni fondamentali relative alla specifica di linguaggi formali e alla traduzione di programmi. Sebbene il corso presenti queste nozioni nel contesto specifico della progettazione e realizzazione di compilatori, i concetti, i formalismi, le metodologie e le tecniche presentate trovano applicazione in un'ampia gamma di contesti in cui si richiede la specifica, la comprensione e la traduzione di informazioni strutturate.

La parte teorica dell'insegnamento è affiancata da attività pratiche mirate a illustrare l'applicazione concreta delle suddette tecniche allo sviluppo di programmi per l'analisi di dati strutturati e di altri programmi.

7. Programma:

- Nozioni di base di linguaggi formali
- Automi a stati finiti
- Espressioni regolari
- Proprietà dei linguaggi regolari
- Grammatiche libere
- Automi a pila
- Proprietà dei linguaggi liberi
- Analisi sintattica top-down
- Traduzione diretta dalla sintassi
- Generazione di codice intermedio

Insegnamento **INF0291 - Logica**

Insegnamento (inglese):	**Logic**
CFU:	**6**
Settore:	**MAT/01 - LOGICA MATEMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Alessandro ANDRETTA (Titolare)** **Luca MOTTO ROS (Titolare)** **Matteo VIALE (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza delle basi della matematica della scuola superiore: in particolare le operazioni aritmetiche di base, le proprietà delle potenze, le equazioni di primo e secondo grado. Conoscenza della terminologia di base relativa alle parti del discorso: nomi, verbi, proposizioni, aggettivi.

Eventuali corsi propedeutici

Nessuno.

2. Obiettivi formativi:

L’insegnamento si propone di fornire allo studente una introduzione alla logica matematica, con particolare riguardo per i suoi aspetti più rilevanti per la formazione di base di un informatico, in particolare una adeguata familiarità con le strutture algebriche e le principali tecniche di dimostrazione.

7. Programma:

Tecniche di dimostrazione (8 ore circa) • Dimostrazione diretta, per assurdo, per contrapposizione; • Connettivi logici e loro significato in termini di condizioni di verità; • Tavole di verità e conseguenza logica tra proposizioni.

Insiemistica di base (8 ore circa) • Insiemi • Relazioni • Funzioni

Cardinalità (10 ore circa) • Insiemi numerabili e più che numerabili: esempi; • Operazioni infinitarie: unioni e intersezioni, prodotti e somme, con le principali proprietà.

Il principio di induzione (6 ore circa) • Forma ordinaria e forte del principio di induzione; • Principio del minimo; • Equivalenza tra forme del principio di induzione; • Induzione strutturale; • Ricorsione.

Formalizzazione (6 ore circa) • Linguaggi proposizionali e del prim’ordine: termini, quantificatori, alfabeto non logico, formule; • Schemi di traduzione da linguaggio naturale in linguaggi del prim’ordine (condizione sufficiente, necessaria, per tutti gli n abbastanza grandi, ci sono n arbitrariamente grandi,...)

Semantica della Logica del primo ordine (14 ore circa) • Regole della semantica di Tarski per formule del primo ordine: esempi, e definizioni; • Conseguenza logica tra formule del primo ordine: definizioni e costruzione di contro-esempi.

Insegnamento **INF0003 - Logica per l'Informatica**

Insegnamento (inglese):	**Logic for Computer Science**
CFU:	**6**
Settore:	**MAT/01 - LOGICA MATEMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Luca Luigi PAOLINI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

- Conoscenze elementari ed informali di logica e teoria degli insiemi - Conoscenze elementari di linguaggi formali

Eventuali corsi propedeutici

- Logica e Matematica Discreta - Linguaggi Formali e Compilatori

2. Obiettivi formativi:

Il corso mira ad offrire agli studenti gli strumenti formali per il ragionamento rigoroso sia nel contesto informale dei linguaggi naturali che in quelli assistiti dal calcolatore. In particolare si vuole enfatizzare le potenzialita' della logica come strumento per una descrizione rigorosa della specifica di un programma e per porre le basi della verifica automatica di proprietà di programmi.

7. Programma:

Logica proposizionale:

- sistemi deduttivi, interpretazione e modelli, correttezza e completezza;
- risoluzione, BDD, SAT solver.

Logica al primo-ordine:

- sistemi deduttivi, interpretazione e modelli, correttezza e completezza;
- resolution and Prolog.

Cenni ad altre sistemi logici.

Insegnamento **INF0290 - Matematica Discreta**

Insegnamento (inglese):	**Discrete Mathematics**
CFU:	**6**
Settore:	**MAT/02 - ALGEBRA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Cristina BERTONE (Titolare)** **Yu CHEN (Titolare)** **Andrea MORI (Titolare)** **Lea TERRACINI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza delle basi della matematica della scuola superiore: in particolare le operazioni aritmetiche di base, le proprietà delle potenze, le equazioni di primo e secondo grado. Conoscenza della terminologia di base relativa alle parti del discorso: nomi, verbi, proposizioni, aggettivi.

Eventuali corsi propedeutici

Nessuno

2. Obiettivi formativi:

L’insegnamento si propone di fornire allo studente una introduzione alla matematica discreta, con particolare riguardo per gli aspetti più rilevanti per la formazione di base di un informatico, in particolare una adeguata familiarità con le strutture algebriche, il calcolo combinatorio e le principali tecniche di dimostrazione.

7. Programma:

Linguaggio degli insiemi (14 ore circa) • Insiemi: insieme vuoto; sottoinsiemi; unione; intersezione; complementare; insieme delle parti (con particolare attenzione al caso finito). • Corrispondenze, relazioni e funzioni: relazioni d’ordine. • Relazioni di equivalenza e partizioni. • Composizione e inversione di corrispondenze. • Iniettività, suriettività, composizione e invertibilità di funzioni.

Calcolo combinatorio (12 ore circa) • Cardinalità di insiemi finiti • Principi della somma e del prodotto • Disposizioni semplici e con ripetizioni • Combinazioni semplici e con ripetizioni. • Il Teorema del binomio e il triangolo di Pascal-Tartaglia • Il principio di inclusione-esclusione

Strutture algebriche (10 ore circa) • Semigruppi e loro morfismi. • Monoidi e loro morfismi: monoide delle parole • Gruppi e loro morfismi. • Alcuni esempi di strutture algebriche: numeri naturali e interi, gruppo delle biiezioni di un insieme. • Gruppi e sottogruppi ciclici. • Sottogruppi e Teorema di Lagrange. • Corpi e campi: campo dei numeri razionali.

Aritmetica modulare (8 ore circa) • Anelli degli interi e delle classi di resto. • Teorema della divisione • L'algoritmo di Euclide • Identità di Bezout • Equazioni diofantee • Il teorema di Eulero-Fermat

Gruppo delle permutazioni (8 ore circa) • Composizione, potenze e inverse di permutazioni. • Decomposizione in cicli disgiunti e decomposizione in trasposizioni. • Parità di una permutazione • Sottogruppi del gruppo delle permutazioni.

Insegnamento	**MFN0633 - Metodi Formali dell'Informatica**

Insegnamento (inglese):	**Formal Methods in Computer Science**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1** **2**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Ugo DE' LIGUORO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Sono richieste conoscenze elementari di logica e buone conoscenze di programmazione e algoritmi; inoltre si assume che lo studente possegga nozioni di linguaggi formali.

Eventuali corsi propedeutici

Matematica Discreta e Logica, Programmazione 1 e 2, Algoritmi e Strutture Dati, Linguaggi Formali e Traduttori.

2. Obiettivi formativi:

Il corso si propone di offrire agli studenti le conoscenze di base relative alla verifica formale di proprietà dei programmi. A questo scopo vengono introdotte le basi teoriche dei sistemi di riscrittura, del lambda calcolo e della teoria dei tipi. Quindi sarà trattato un semplice linguaggio imperativo, di cui verranno esposti la semantica operazionale e quella logica, basata sulla logica di Hoare.

7. Programma:

- Metodi formali
  - Logica equazionale e riscrittura
  - Deduzione naturale
  - Lambda calcolo
- Il sistema Agda
  - Funzioni e tipi di dato
  - Logica costruttiva
  - Liste e strutture dati
  - Verifica di programmi funzionali
- Fondamenti di linguaggi di programmazione procedurali
  - Sintassi del linguaggio IMP
  - Semantica operazionale big-step e small-step
  - Sistemi di tipo per IMP
  - Logica di Hoare
  - Completezza relativa
  - Verification conditions
  - Separation logic

Insegnamento	**INF0190 - Metodi Formali dell'Informatica - Parte A**

Insegnamento (inglese):	**Formal Methods in Computer Science - A**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1** **2**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Ugo DE' LIGUORO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Sono richieste conoscenze elementari di logica e buone conoscenze di programmazione e algoritmi; inoltre si assume che lo studente possegga nozioni di linguaggi formali.

Eventuali corsi propedeutici

Matematica Discreta e Logica, Programmazione 1 e 2, Algoritmi e Strutture Dati, Linguaggi Formali e Traduttori.

2. Obiettivi formativi:

Il corso si propone di offrire agli studenti le conoscenze di base relative alla verifica formale di proprietà dei programmi. A questo scopo vengono introdotte le basi teoriche dei sistemi di riscrittura, del lambda calcolo e della teoria dei tipi. Quindi sarà trattato un semplice linguaggio imperativo, di cui verranno esposti la semantica operazionale e quella logica, basata sulla logica di Hoare.

7. Programma:

- Metodi formali
  - Logica equazionale e riscrittura
  - Deduzione naturale
  - Lambda calcolo
- Il sistema Agda
  - Funzioni e tipi di dato
  - Logica costruttiva
- Fondamenti di linguaggi di programmazione procedurali
  - Sintassi del linguaggio IMP
  - Semantica operazionale big-step e small-step
  - Logica di Hoare
  - Completezza relativa
  - Verification conditions

Insegnamento	**INF0193 - Metodologie e Tecnologie Didattiche per l'Informatica (PF24)**

Insegnamento (inglese):	**Methodologies and technologies for teaching informatics (PF24)**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Sara CAPECCHI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Basi di programmazione, conoscenza degli algoritmi fondamentali, dimestichezza con con strumenti di modellazione concettuale.

2. Obiettivi formativi:

Il corso mira a:

1. - Fornire la conoscenza di alcuni aspetti dell'Informatica come disciplina scientifica e delle motivazioni alla base della necessità del suo insegnamento;
2. -Favorire comprensione di aspetti pedagogici e teorie dell'apprendimento nel contesto dell'insegnamento dell'Informatica;
3. - Sviluppare capacità riflessive sulla disciplina, facendo emergere per confrontare le idee pregresse degli studenti sugli argomenti trattati.
4. - Introdurre delle principali metodologie per la costruzione di un curriculum di Informatica coerente con gli obiettivi fissati dalle indicazioni nazionali e dalle linee guida.
5. - Sperimentare attività laboratoriali per lo sviluppo e la valutazione di competenze e riconoscimento della loro importanza
6. - Acquisire pratiche didattiche e di processi di insegnamento e apprendimento dell'Informatica sia con l'uso delle tecnologie digitali sia con tecniche di tipo csunplugged ovvero senza calcolatore con attività per i diversi livelli di scuola.
7. - Riconoscere la relazione esistente tra l'Informatica e temi rilevanti per la società quali l'etica dei materiali.

7. Programma:

1. -Analisi delle principali metodologie per l'insegnamento dell'informatica presenti in letteratura.

1. -Problem solving: dalla formulazione del problema alla progettazione della soluzione

1. -Processi di insegnamento e apprendimento dell'Informatica con e senza l'uso delle tecnologie. In particolare analisi di metodologie e tecniche didattiche quali:

-csunplugged ovvero didattica dell'informatica senza calcolatore -costruttivismo, costruzionismo.

1. -Didattica della programmazione.

Insegnamento **MFN0582 - Programmazione I**

Insegnamento (inglese):	**Programming I**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Cristina BAROGLIO (Esercitatore)** **Valerio BASILE (Titolare)** **Marco BECCUTI (Titolare)** **Marco BECCUTI (Esercitatore)** **Robert RenÃ¨ Maria BIRKE (Titolare)** **Felice CARDONE (Titolare)** **Luigi DI CARO (Titolare)** **Attilio FIANDROTTI (Titolare)** **Alessandro MAZZEI (Titolare)** **Luca ROVERSI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Non è richiesto alcun prerequisito specifico alla programmazione. È consigliabile avere capacità di uso del calcolatore con sistema a finestre. È invece opportuno possedere conoscenze di base, quali i concetti di numero (naturale, intero, razionale, reale), di funzione, le quattro operazioni, elevamento a potenza, radice, esponenziale, logaritmo, piano cartesiano, calcolo algebrico elementare. Infine, è indispensabile una buona padronanza della lingua madre, possibilmente accompagnata da una propensione al ragionamento strutturato.

Eventuali corsi propedeutici

Nessuno.

2. Obiettivi formativi:

Questo insegnamento concorre agli obiettivi formativi di un Corso di Laurea in Informatica, mirando a fornire le conoscenze di base della programmazione e del funzionamento di un interprete virtuale, propedeutici alla quasi totalità dei corsi a seguire.

7. Programma:

- Struttura di base di un calcolatore;
- Informazioni essenziali su linguaggi di programmazione, differenze tra compilatori e interpreti;
- Algoritmi iterativi e ricorsivi;
- Linguaggio di riferimento: variabili, tipi di dato fondamentali e array, assegnazione e controllo del flusso, procedure e funzioni con parametri, modello di gestione della memoria;
- Correttezza parziale, terminazione e intuizioni sulla nozione di costo di un algoritmo.

Insegnamento **MFN0585 - Programmazione II**

Insegnamento (inglese):	**Programming II**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**A - di base**
Docenti:	**Stefano BERARDI (Titolare)** **Viviana BONO (Titolare)** **Fabio CIRAVEGNA (Titolare)** **Ferruccio DAMIANI (Titolare)** **Ferruccio DAMIANI (Esercitatore)** **Diego MAGRO (Titolare)** **Gianluca TORTA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Agli studenti è richiesta la conoscenza delle nozioni fondamentali di programmazione imperativa nel linguaggio Java (comandi di assegnamento, condizionali e iterativi, tipi di dato semplici e array, nozioni di astrazione procedurale e ricorsione).

Eventuali corsi propedeutici

Programmazione I

2. Obiettivi formativi:

L’insegnamento ha l'obiettivo di approfondire concetti di informatica di base e in particolare di fornire una introduzione al paradigma di programmazione a oggetti. Si propone quindi di raffinare le capacità di programmare nel linguaggio Java apprese nel corso di Programmazione I e di introdurre le nozioni fondamentali della programmazione orientata agli oggetti. In particolare, il corso illustrerà le *astrazioni fondamentali* per la progettazione del software (classi e oggetti), la definizione di *semplici strutture dati* (liste, alberi, pile, code) e operazioni corrispondenti, i meccanismi di base per favorire *riuso e modularità* del software (ereditarietà, polimorfismo, tipi generici), la specifica degli *invarianti di classe* e gestione delle loro violazioni (asserzioni ed eccezioni), così come alcune *classi fondamentali* della libreria Java. Si darà particolare enfasi agli aspetti di buona progettazione del software, utilizzando concetti presi a prestito dall'ingegneria del software e formalismi grafici quali UML.

7. Programma:

Il programma del corso è basato sul testo di riferimento, eventualmente integrato da dispense e/o altro materiale fornito dal docente e reso disponibile sulla pagina Moodle del corso. Segue un sommario degli argomenti trattati e dei capitoli relativi sul testo di riferimento. Tale sommario non riflette necessariamente l’ordine di esposizione degli argomenti.

- Ripasso dei concetti di base della programmazione imperativa e di Java (Cap. 1-7).
- Incapsulamento, definizione di una classe, creazione e uso di oggetti (Cap. 8 e 9).
- Specializzazione di una classe, ereditarietà (Cap. 10)
- Relazioni tra classi, polimorfismo, classi parzialmente definite, interfacce (Cap. 11).
- Generalizzazione di una classe, classi parametriche, tipi generici (Cap. 12).
- Invarianti di classe, asserzioni, eccezioni e loro gestione (Cap. 13)
- progettazione e implementazione di strutture dati (liste, alberi, pile, code) (Cap. 15).
- Collezioni e iteratori (Cap. 16).

- Cenni su progettazione a oggetti e UML.

Insegnamento **MFN0605 - Programmazione III**

Insegnamento (inglese):	**Programming III**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Liliana ARDISSONO (Titolare)** **Roberto ESPOSITO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si richiede una buona conoscenza dell'analisi e della progettazione object-oriented (fornita dagli insegnamenti Programmazione II e Algoritmi e strutture dati) e dei meccanismi di base e delle problematiche della programmazione concorrente (fornita dall'insegnamento Sistemi Operativi). Gli studenti e le studentesse devono inoltre avere la capacità di scrivere, compilare e verificare la correttezza di programmi in Java.

Eventuali corsi propedeutici

Programmazione II, Algoritmi e strutture dati, Sistemi Operativi.

2. Obiettivi formativi:

Lo sviluppo di software efficiente e scalabile presuppone la capacità di programmare applicazioni distribuite e concorrenti. In particolare, la programmazione distribuita in ambiente object oriented arricchisce le nozioni di base di programmazione concorrente sfruttando il paradigma ad oggetti per una più chiara scomposizione delle attività da eseguire in parallelo e loro attribuzione alle entità software di competenza, che possono essere modellate come oggetti distribuiti che offrono i relativi servizi.

L'insegnamento si pone l’obiettivo di fornire la conoscenza di base necessaria per la programmazione di applicazioni object-oriented distribuite e concorrenti, usando un linguaggio ad alto livello (Java), attraverso (i) lo scambio di dati in rete, e (ii) la programmazione di thread paralleli, cioè di processi “leggeri” che possano operare su uno o più processori all’interno della stessa applicazione principale. Altro obiettivo fondamentale dell'insegnamento è la tecnica di programmazione ad eventi per la realizzazione di interfacce grafiche, che stanno alla base di tutte le applicazioni desktop e web basate su finestre.

In questo insegnamento, le capacità di applicare le conoscenze acquisite vengono sviluppate nell'ambito di esercitazioni sperimentali che permettono di applicare le nozioni apprese durante le ore di teoria, organizzando il lavoro in piccoli gruppi. I temi introdotti durante le sperimentazioni corredano e integrano le conoscenze derivanti dalla parte teorica (knowledge and understanding) e permettono agli studenti e alle studentesse di familiarizzare con le metodologie e tecnologie introdotte, e di analizzare soluzioni alternative (applying knowledge and understanding). Inoltre nell'ambito delle attività sperimentali è previsto lo sviluppo di un’applicazione SW distribuita realistica (denotata come PROGETTO SW), con interfaccia grafica, che va oltre la complessità del tipico esercizio svolto durante le ore di lezione. Lo svolgimento del progetto SW permette di allenare l'autonomia di giuzio nella scelta delle implentazioni da realizzare, analizzando in forma critica le informazioni acquisite nella parte teorica dell'insegnamento. La preparazione del progetto SW è anche volta a stimolare la capacità comunicativa degli studenti e delle studentesse che sono invitati/e a illustrare verbalmente le soluzioni adottate (communication skills).

7. Programma:

Programmazione ad eventi in Java: programmare interfacce grafiche (SWING e JavaFX/FXML)
- Sorgenti di eventi, gestori di eventi, event-driven programming
- Organizzazione e uso delle interfacce grafiche di Java
- L'architettura Model-View-Controller (MVC)

Programmazione Multithread:
- Esecuzione concorrente di istruzioni
- I Thread in Java: ciclo di vita dei Thread
- Creazione e sincronizzazione di thread
- Estensione del modello della memoria in presenza di Thread
- Problemi di sincronizzazione e loro risoluzione mediante il linguaggio Java

Programmazione in rete in Java:
- L'architettura client-server
- Java socket
- Polimorfismo e trasferimento di oggetti mediante Java
- Il modello di esecuzione distribuita di oggetti

Insegnamento **MFN0635 - Reti di Elaboratori**

Insegnamento (inglese):	**Computer Networks**
CFU:	**12**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1** **2**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Michele GARETTO (Titolare)** **Matteo SERENO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

- Competenza nella programmazione sequenziale C e/o Java (per recepire gli elementi di programmazione concorrente e distribuita presentati nel corso);
- Competenza nelle architetture degli elaboratori e dei sistemi operativi (per apprezzare gli aspetti sistemistici dei sistemi distribuiti oggetto di studio).

Eventuali corsi propedeutici

Programmazione I e II, Architetture, Sistemi Operativi.

2. Obiettivi formativi:

Il corso studia gli elementi fondamentali delle tecnologie di trasmissione del livello data link, dei protocolli di accesso a mezzi condivisi e dei protocolli di trasmissione wireless, la suite di protocolli TCP/IP, e i principi che guidano la strutturazione e la progettazione di applicazioni distribuite.

7. Programma:

- Introduzione ad Internet ed alle reti di calcolatori
- Il livello Applicazioni
- Il livello di Trasporto
- Il livello di Rete
- Il livello di collegamento e le reti locali: collegamenti, reti di accesso e reti locali
- Reti Wireless e Mobili

Insegnamento **MFN1362 - Reti I**

Insegnamento (inglese):	**Computer Networks I**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Marco BOTTA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Lo studente deve conoscere prima di seguire il corso i fondamenti della programmazione e dei sistemi operativi.

Eventuali corsi propedeutici

Sistemi Operativi, Programmazione I e II

2. Obiettivi formativi:

Il corso si propone di fornire agli studenti nozioni base sulle reti di calcolatori e una comprensione approfondita della suite di protocolli TCP/IP. Inoltre, attraverso l'uso di software ed esempi pratici, il corso fornisce agli studenti una comprensione concreta dei meccanismi di comunicazione tra dispositivi di rete e calcolatori.

7. Programma:

Fondamenti su reti di calcolatori.

Parte introduttiva alle reti (i livello di comunicazione fisico, il livello link, reti wireless e mobilità) Il livello rete: instradamento e reti IP Il livello transport: controllo della congestione e del flusso end-to-end Il livello applicativo: suite di applicazioni TCP/IP (Web, Posta, DNS, File transfer, sistemi P2P).

Insegnamento **INF0002 - Servizi Web**

Insegnamento (inglese):	**Web Services**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Liliana ARDISSONO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si richiede una buona conoscenza delle basi di dati (fornita dall'insegnamento Basi di Dati), dei sistemi operativi (fornita dall'insegnamento Sistemi Operativi), dell'analisi e della progettazione object-oriented (fornita dagli insegnamenti Programmazione II, e Algoritmi e strutture dati) e dei fondamenti della programmazione distribuita (fornita dall'insegnamento Programmazione III). Gli studenti e le studentesse devono inoltre avere la capacità di scrivere, compilare e verificare la correttezza di programmi in Java.

Eventuali corsi propedeutici

Basi di Dati, Sistemi Operativi, Interazione Uomo-Macchina, Programmazione II, Algoritmi e Strutture Dati, Programmazione III.

2. Obiettivi formativi:

L’implementazione delle applicazioni web richiede di guardare “dietro all’interfaccia utente” per andare a fondo su aspetti architetturali e tecnologici che possono influenzare non solo le prestazioni, la scalabilità e la robustezza, ma anche le tipologie di servizio che possono essere effettivamente offerte.

L'insegnamento si pone come obiettivo di fornire la conoscenza di base necessaria per la progettazione e lo sviluppo di applicazioni Web interattive, accessibili da terminali desktop e mobili (grazie all’uso di linguaggi di interfaccia utente cross-platform), e caratterizzate da una logica applicativa mediamente complessa. Ci si propone di formare programmatori/programmatrici capaci di sviluppare applicazioni web di qualità e basate su architetture standard, largamente utilizzate nel mondo aziendale.

L'insegnamento tratterà dal punto di vista sia teorico che pratico: (i) lo sviluppo di pagine web statiche e dinamiche responsive; (ii) la programmazione lato server. Inoltre, l'insegnamento tratterà la rappresentazione e interpretazione dei dati in XML, data l'importanza nella gestione dell'interoperabilità tra applicazioni.

In questo insegnamento, le capacità di applicare le conoscenze acquisite vengono sviluppate nell'ambito di esercitazioni sperimentali che permettono di applicare le nozioni apprese durante le lezioni teoriche, organizzando il lavoro in piccoli gruppi. I temi introdotti durante le esercitazioni corredano e integrano le conoscenze derivanti dalla parte teorica (knowledge and understanding) e permettono agli studenti e alle studentesse di familiarizzare con le metodologie e tecnologie introdotte, anche investigando soluzioni alternative (applying knowledge and understanding). Inoltre essi/esse sviluppano un’applicazione SW realistica con interfaccia web (denotata come PROGETTO SW), che va oltre la complessità del tipico esercizio svolto durante le esercitazioni. La progettazione e lo sviluppo del progetto SW permettono di allenare l'autonomia di giuzio nella scelta delle implentazioni da realizzare, analizzando in forma critica le informazioni acquisite nella parte teorica dell'insegnamento. La preparazione del progetto SW è anche volta a stimolare la capacità comunicativa degli studenti e delle studentesse che sono invitati/e a illustrare verbalmente le soluzioni adottate (communication skills).

7. Programma:

Architetture delle applicazioni Web: Web browser e Web server.

Il Pattern Model View Controller per le applicazioni Web - Progettazione e sviluppo di applicazioni Web a 3 livelli basate su MVC:

- Il primo livello (client dell'applicazione): HTML5, CSS, scripting lato client (JavaScript e AJAX). Raccolta dati con HTML form.
- Il secondo livello (logica applicativa): Servlet Java.
- Il terzo livello (livello dei dati): accesso a database relazionali con Java Database Connectivity (JDBC), rappresentazione e gestione di informazioni con XML (XML Schema, XPath, XML Parsers).
- Framework le applicazioni web MVC (Vue.js).
- Introduzione al linguaggio PHP per lo sviluppo di pagine web dinamiche.

Insegnamento	**MFN0636 - Sicurezza**

Insegnamento (inglese):	**Computer and Network Security**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**B - caratterizzante** **D - libera**
Docenti:	**Francesco BERGADANO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Si presuppone la conoscenza dei sistemi operativi e delle reti di calcolatori basate sui protocolli della suite TCP/IP. La conoscenza di specifiche piattaforme di sviluppo e programmazione, ed in particolare il linguaggio C, è considerata di aiuto alla comprensione di alcuni argomenti svolti nel corso.

Eventuali corsi propedeutici

Sistemi Operativi, Reti di Calcolatori, Programmazione 1 e 2

2. Obiettivi formativi:

Il corso si propone di fornire agli studenti gli strumenti crittografici e tecnici utilizzati per garantire la sicurezza del software, delle reti e dei calcolatori.

In particolare il corso si articola in due parti: (1) crittografia e (2) sicurezza del software e delle reti, ciascuna con diversi obiettivi formativi.

Nella prima parte l'obiettivo formativo è principalmente orientato a far comprendere nel dettaglio alcuni esempi di cifrari, sia simmetrici che asimmetrici, ed alcuni metodi di autenticazione. Nella seconda parte l'obiettivo consiste nell'illustrare i principali attacchi alla sicurezza del software (es. memory corruption, injection) e delle reti (es. sniffing, spoofing, DoS), illustrando le possibili contromisure.

Inoltre, attraverso l'uso di esempi pratici, il corso fornisce agli studenti una comprensione concreta dei maggiori rischi di sicurezza e delle soluzioni disponibili. Tali rischi vengono analizzati seguendo le più diffuse best practice e con riferimento agli standard internazionali del settore.

7. Programma:

Strumenti crittografici: cifrari simmetrici e asimmetrici, funzioni di hash resistenti alle collisioni, attacchi del compleanno, autenticazione simmetrica e firma elettronica

Sicurezza delle reti e del software: sicurezza delle reti locali e geografiche, disponibilità e Denial of Service, sicurezza perimetrale e firewall, sicurezza del software (memory corruption, Web Application security) reti private virtuali, analisi dei rischi di cybersecurity

Insegnamento	**MFN0618 - Sistemi Informativi**

Insegnamento (inglese):	**Information Systems**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Roberto MICALIZIO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

E' richiesta la conoscenza dei concetti relativi alle basi di dati relazionali e alla modellazione concettuale (Entity-Relationship), alla programmazione e allo studio degli algoritmi.

Eventuali corsi propedeutici

Basi Dati .

2. Obiettivi formativi:

L'insegnamento di Sistemi Informativi ha lo scopo di dare una panoramica delle maggiori e più diffuse applicazioni delle basi di dati nel mondo del lavoro e dell'impresa in cui buona parte dei processi aziendali sono ormai automatizzati. A tal fine, il corso introdurrà nozioni basilari di Business Process Management (BPM) e consentirà di sperimentare la modellazione di processi di business mediante lo standard Business Process Modeling Notation (BPMN).
Inoltre, l'insegnamento introduce i sistemi di pianificazione delle risorse aziendali (Enterprise Resource Planning - ERP), i sistemi integrati di gestione e profilazione del cliente (CRM), i sistemi di supporto alla decisione (DSS) e alle piattaforme di Business Intelligence. In particolare, l'insegnamento si focalizzerà sulle metodologie di progettazione delle Data Warehouse e le metodologie di analisi on-line dei dati (OLAP).

7. Programma:

1. Introduzione ai sistemi informativi aziendali (Modello organizzativo, modello funzionale, modello informatico)

2. Sistemi ERP (Le suite ERP; Paradigma ERP; Piattaforme software; Offerta ERP; Trasformazione dell’impresa)

3. Integrazione con il cliente: i sistemi CRM (Ruolo dei sistemi CRM nelle aziende; Schema architetturale; Il paradigma; Esempi; Suite di package software; Evoluzione dei CRM)

4. Piattaforme di Business Intelligence e DSS (Data Warehouse): Livello delle fonti ed ETL; Progettazione del sistema di warehousing; Livello di elaborazione: reporting e DSS; cenni di data mining

5. Gestione dell'azienda orientata ai processi (Studio e sperimentazione di diversi linguaggi di modellazione quali: UML,e BPMN)

Insegnamento **MFN0607 - Sistemi Intelligenti**

Insegnamento (inglese):	**Intelligent Systems**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**B - caratterizzante** **D - libera**
Docenti:	**Cristina BAROGLIO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Poiché Sistemi Intelligenti è il primo corso che tratta argomenti di Intelligenza Artificiale, le competenze attese in ingresso riguardano competenze nel settore informatico. In particolare: - conoscenza di algoritmi su alberi e grafi con relative nozioni di complessità - esperienza di programmazione con particolare riferimento a programmazione ad oggetti (organizzazione in classi e sottoclassi, ereditarietà) - nozioni di logica (calcolo proposizionale e calcolo dei predicati del primo ordine) - nozioni di modelli semantici dei dati nelle basi dati.

Eventuali corsi propedeutici

Gli studenti che sono iscritti al corso di laurea di Informatica di Torino acquisiscono le competenze in ingresso sopra elencate seguendo gli insegnamenti di: - “Algoritmi e strutture dati”, - “Programmazione I e II”, - “Basi di dati”, - “Matematica Discreta e Logica” (e sostenendo i relativi esami)

2. Obiettivi formativi:

Il corso si propone di fornire una introduzione generale alle problematiche nel settore dell'Intelligenza Artificiale, con particolare attenzione a come sia possibile costruire un sistema dotato di capacità autonome di risoluzione di problemi, di ragionamento e di apprendimento quando abbia a disposizione una rappresentazione simbolica del mondo.

Il corso si articola in tre parti principali: - Risoluzione automatica di problemi - Rappresentazione della conoscenza e ragionamento - Nozione di agente intelligente che agisce, ragiona ed apprende

Data la natura introduttiva del corso e la durata del corso, molte problematiche avanzate di Intelligenza Artificiale trovano collocazione nei corsi offerti per l'indirizzo "Sistemi per il Trattamento dell'Informazione" della laurea magistrale in Informatica

7. Programma:

Come già detto l'insegnamemnto è una introduzione ai concetti basilari di Intelligenza articiaicile e si articola in tre parti strettamente connesse.

Parte 1) RISOLUZIONE AUTOMATICA DI PROBLEMI In questa parte si affronta la problematica di come definire il concetto di problema e di soluzione, di distinguere tra soluzione e soluzione ottima. Sono studiati tre approcci alla risoluzione di problemi: ricerca nello spazio degli stati, ricerca in spazi con avversario (giochi ad informazione completa), risoluzione di problemi mediante soddisfacimento di vincoli. Per ciascun approccio si discutono le principali strategie di ricerca: ampiezza, profondità, iterative deepening (per le ricerche cieche nello spazio degli stati), A* e Recursive Best First Strategy (per le ricerche euristiche), Min-Max e Alfa-beta (per i giochi con avversario), backtracking, forward propagation e arc consistency per meccanismi basati su soddisfacimento di vincoli. Particolare attenzione viene data alla garanzie offerte dalle diverse strategie in termini di qualità della soluzione e di complessità computazionale.

Parte 2) RAPPRESENTAZIONE DELLA CONOSCENZA E RAGIONAMENTO Il problema della rappresentazione della conoscenza e dei relativi meccanismi inferenziali viene affrontato studiando due principali famiglie di approcci alla rappresentazione della conoscenza: formalismi logici e rappresentazioni strutturate. Per quanto riguarda i formalismi logici si vede come sia il calcolo proposizionale che il calcolo dei predicati del primo ordine possano essere utilizzati per rappresentare conoscenza sul mondo e si vede come i meccanismi inferenziali (modus ponens, resolution, etc.) possano essere adoperati per fornire servizi utili (es. risposta a domande, verifica consistenza, ecc.). Si analizza anche come una rappresentazione a regole permetta meccanismi di ragionamento più efficienti (forward e backward chaining). Notevole attenzione viene data alla rappresentazione della conoscenza strutturata introducendo tassonomie, classi, individui, ereditarietà singola e multipla, inferenze specializzate. Queste nozioni vengono analizzate ed esemplificate mediante uso del linguaggio ontologico OWL2 (proposto e supportato da W3C).

Parte 3) AGENTI E APPRENDIMENTO AUTOMATICO In questa parte conclusiva si introduce la nozione di agente intelligente che opera in un ambiente e si fa vedere come l’agente possa avere sia comportamenti reattivi che deliberativi a seconda del compito assegnato. Si illustra come agente debba avere capacità di risoluzione automatica di problemi e di ragionamento sullo stato del mondo e sul suo stato. Si descrive brevemente come l’apprendimento automatico sia una delle caratteristiche essenziali per ottenere un agente intelligente. Vengono introdotte solo nozioni elementari con particolare riguardo all'apprendimento da esempi (in particolare apprendimento di alberi di decisione). Viene infine fatta una introduzione alle reti neuronali come strumento per passare dal livello sub simbolico a quello simbolico.

Insegnamento **MFN0601 - Sistemi Operativi**

Insegnamento (inglese):	**Operating Systems**
CFU:	**12**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Cristina BAROGLIO (Titolare)** **Enrico BINI (Titolare)** **Massimiliano DE PIERRO (Titolare)** **Daniele GUNETTI (Titolare)** **Daniele Paolo RADICIONI (Titolare)** **Claudio SCHIFANELLA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Agli studenti è richiesta una conoscenza di base dell'architettura di un computer (secondo quanto studiato nel corso di Architetture degli Elaboratori I) e dei concetti di base di programmazione (secondo quanto studiato nel corso di Programmazione I). Gli studenti dovranno inoltre essere in grado di padroneggiare i sistemi di enumerazione binario (base due) ed esadecimale (base sedici).

Eventuali corsi propedeutici

Costituiscono prerequisiti i contenuti dei corsi di:

* Architettura degli Elaboratori I * Programmazione I

2. Obiettivi formativi:

Il sistema operativo costituisce l'interfaccia fondamentale tra l'utilizzatore di un computer e il computer stesso. Parte essenziale del curriculum di base di un laureato in informatica è la conoscenza di come il sistema operativo sia in grado di amministrare le varie componenti hardware di cui è composto un computer. Queste modalità di amministrazione devono essere il più possibile trasparenti al generico utilizzatore del computer, ma devono essere conosciute a fondo da ogni specialista del settore. L'insegnamento fornisce dunque una conoscenza di base dell'architettura interna e del funzionamento dei moderni sistemi operativi, e di come, ai fini di garantire un ragionevole compromesso tra efficienza, sicurezza e facilità d'uso, vengono amministrate le risorse fondamentali della macchina su cui il sistema operativo è installato: il processore, la memoria principale e la memoria secondaria.

Per la parte di laboratorio gli obiettivi formativi sono l'apprendimento del linguaggio C, utilizzato per la programmazione nell'ambiente del sistema operativo Unix. La parte di laboratorio mira a fornire allo studente una conoscenza (teorica e pratica) di base sui comandi della shell, sulla gestione dei processi, sugli strumenti di inter-process communication e sulla gestione dei segnali forniti dal sistema, oltre che alcuni rudimenti di programmazione bash.

I temi introdotti durante il laboratorio corredano e integrano le conoscenze derivanti dalla parte teorica (knowledge and understanding), al tempo stesso presentando esempi di problemi realistici di comunicazione e sincronizzazione su cui gli studenti sono sollecitati a cimentarsi, anche investigando soluzioni alternative (applying knowledge and understanding). La preparazione e la discussione del progetto sono inoltre volte a stimolare le capacità di organizzare il lavoro in piccoli gruppi (2-3 studenti), e poi di illustrare verbalmente le soluzioni adottate (communication skills).

TUTTE LE INFORMAZIONI SUL CORSO E IL MATERIALE DIDATTICO SI TROVERANNO IN MOODLE

7. Programma:

NOTA: Par la parte di teoria, il programma è basato sul TESTO DI RIFERIMENTO.

PARTE DI TEORIA:

* Introduzione al Corso di Sistemi Operativi

* PARTE I: GENERALITA'

o Introduzione (cap. 1)

o Strutture dei Sistemi Operativi (cap. 2)

* PARTE II: GESTIONE DEI PROCESSI

o Processi (cap. 3)

o Thread (cap. 4)

o Scheduling della CPU (cap. 5)

o Sincronizzazione dei Processi (cap. 6)

o Deadlock (Stallo di Processi) (cap. 7)

* PARTE III: GESTIONE DELLA MEMORIA (PRIMARIA)

o Memoria Centrale (cap. 8)

o Memoria Virtuale (cap. 9)

* PARTE IV: GESTIONE DELLA MEMORIA SECONDARIA

o Interfaccia del File System (cap. 10)

o Realizzazione del File System (cap. 11)

o Memoria Secondaria e Terziaria (Gestione dell'Hard disk)

==========

PARTE DI LABORATORIO:

- Linguaggio C
- Introduzione a Unix (comandi, shell, file system, diritti d'accesso, ridirezione, pipe)
- Make e makefile
- System call per la creazione e la sincronizzazione di processi
- System call per L'InterProcess Communication e per la gestione di segnali
- Esercitazioni pratiche, in particolare: esercitazioni finalizzate ad imparare il linguaggio C, ad utilizzare Unix e a sviluppare programmi concorrenti

Insegnamento **INF0004 - Storia dell'Informatica**

Insegnamento (inglese):	**History of Computer Science**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1** **2**
Tipologia di Attività Formativa:	**D - libera**
Docenti:	**Felice CARDONE (Titolare)** **Daniele GUNETTI (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Conoscenza di base dell'architettura di un computer e di un sistema operativo. Dimestichezza di base con i principali paradigmi di programmazione.

Eventuali corsi propedeutici

Architetture degli elaboratori I. Programmazione I e II. Sistemi Operativi.

2. Obiettivi formativi:

Dare una visione di insieme e una prospettiva storica dei momenti salienti dell'evoluzione delle idee teoriche e pratiche dell'informatica.

7. Programma:

Il corso fornisce una introduzione alla storia dell’informatica, con particolare attenzione ad alcune tappe che hanno fornito un contributo essenziale alla definizione della forma attuale dell’informatica (per esempio, lo sviluppo di Internet con il suo retroterra culturale e tecnologico). L’interesse centrale del corso è per le idee e la loro evoluzione, più che per le singole innovazioni tecnologiche: non si tratta di una rassegna di modelli di macchina calcolatrice, ma di una introduzione ai modi di pensiero, alle metafore ed alle visioni che hanno caratterizzato la scienza dell’informazione e del calcolo attraverso la storia.

La prima parte del corso sarà dedicata ad una storia dell’evoluzione delle architetture, dei sistemi operativi e dei linguaggi di programmazione, insieme ad alcuni cenni di storia dell’informatica commerciale e di Internet.

La seconda parte consisterà di lezioni monografiche dedicate alla lettura di lavori classici dell’informatica che hanno contribuito a formarne l’impostazione attuale. Questi lavori saranno resi disponibili sulla pagina Moodle all’inizio della seconda parte del corso.

L’esame del corso prevede un compito finale il cui scopo sarà quello di verificare la comprensione della portata tecnologica e culturale degli eventi salienti della storia dell’informatica.

AVVISO per gli studenti del DAMS che hanno in piano di studi l'insegnamento di Fondamenti di Informatica II, che quest'anno mutua da Storia dell'informatica. Si intende che la parte mutuata è la seconda, che sarà tenuta nel secondo semestre dal Prof. Cardone.

Insegnamento **MFN0606 - Sviluppo delle Applicazioni Software**

Insegnamento (inglese):	**Development of Software Applications**
CFU:	**9**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**2**
Tipologia di Attività Formativa:	**B - caratterizzante**
Docenti:	**Matteo BALDONI (Titolare)** **Robert RenÃ¨ Maria BIRKE (Titolare)** **Sara CAPECCHI (Titolare)** **Ferruccio DAMIANI (Titolare)** **Claudia PICARDI (Titolare)** **Mirko POLATO (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Il corso assume le seguenti conoscenze:
- familiarità con le tecniche di programmazione in linguaggi di alto livello come Java;
- conoscenza dei concetti di base di progettazione e interrogazione di basi di dati.

Eventuali corsi propedeutici

Le competenze del primo punto sono fornite dai corsi di programmazione della laurea triennale, come i corsi di Programmazione I e Programmazione II. Le competenze del secondo punto da un corso di introduzione alle basi di dati, come il corso di Basi di Dati.

2. Obiettivi formativi:

L'insegnamento concorre al raggiungimento degli obiettivi formativi del Corso di Laurea in Informatica fornendo i concetti di base dell'ingegneria del software e allo sviluppo di applicazioni software, utilizzando la metodologia Agile Unified Process (UP), che sfrutta il linguaggio di modellazione UML. Si propone perciò di fornire le competenze teoriche e pratiche necessarie per sviluppare un'applicazione significativa individuando con chiarezza la logica applicativa, l'interazione con le basi di dati e le interfacce richieste dai requisiti. Inoltre dovrà imparare a pianificare il lavoro secondo i canoni dello sviluppo dei progetti: lavoro di gruppo, definizione degli obiettivi e delle fasi di sviluppo.

7. Programma:

1) Elementi di ingegneria del software: modelli Waterfall, Spirale, V-shaped, Componend-based Development, metodologie Agili.

2) Introduzione all'UML: use case diagram, class diagram, object diagram, sequence diagram, communication diagram, state chart, activity diagram.

3) Una metodologia Agile: Unified Process (UP). Fasi di sviluppo (ideazione, elaborazione, costruzione). Casi d'uso, modello di dominio, diagrammi di Sequenza di Sistema, Contratti, Architettura logica e organizzazione in layer, Progettazione (DSD e DCD), GRASP, GoF.

4) Testing: unit testing, acceptance test, white e black box testing.

Insegnamento **MFN0634 - Tecnologie Web**

Insegnamento (inglese):	**Web Technologies**
CFU:	**6**
Settore:	**INF/01 - INFORMATICA**
Periodo didattico:	**1**
Tipologia di Attività Formativa:	**B - caratterizzante** **D - libera**
Docenti:	**Marco BOTTA (Titolare)**

1. Prerequisiti e Propedeuticità:

Competenze attese in ingresso

Lo studente deve mostrare di possedere una buona familiarità con i principi della progammazione (imperativa, ad oggetti e basata su eventi). Inoltre, deve conoscere le basi operative per gestire una base di dati basata su SQL e per configurare/installare pacchetti software nel proprio sistema operativo.

Eventuali corsi propedeutici

- MFN0582 - Programmazione I
- MFN0585 - Programmazione II
- MFN0601 - Sistemi Operativi
- MFN0602 - Basi di Dati

2. Obiettivi formativi:

Il corso si inserisce all'interno del percorso di studi fornendo allo studente le nozioni necessarie per lo sviluppo di applicazioni web. In particolare, gli obiettivi di questo corso sono i seguenti:

- Imparare a produrre siti Web dinamici, animati, interattivi e collegati ad un database in back end;
- Imparare diversi linguaggi e tecnologie per lo sviluppo Web client-side, quali HTML5, CSS, JavaScript, JQuery
- Imparare principi della programmazione server side tramite principalmente PHP e MySQL, sfruttando strumenti opensource come i comuni browser web e il server web Apache.

7. Programma:

- Progettazione base ed implementazione di siti Web
- Presentazione delle diverse strategie di navigazione e di organizzazione dei siti
- Tecnologie client-side, tra cui HTML5, CSS, Javascript, JSON e JQuery
- Tecnologie server side, facendo particolare attenzione alle implementazioni in PhP
- Gestione dei dati in back end
- Tecnologie emergenti (MVC, bootstrap, angular, versioning con github, etc.)