

Guida dello Studente

Corso di Laurea in Informatica

della Facoltà di Scienze MM. FF. NN.

A.A. 2004-2005

A partire dall'anno accademico 2002-2003 è istituito presso l'Università di Parma il **Corso di Laurea in Informatica**, appartenente alla Classe XXVI delle Lauree universitarie in *Scienze e Tecnologie Informatiche* ed afferente alla Facoltà di Scienze MM. FF. NN.

Perché scegliere il Corso di Laurea in Informatica

L'informatica è la scienza che studia l'*elaborazione automatica dell'informazione*, affrontando le problematiche connesse alla sua rappresentazione, manipolazione e trasmissione.

È stato più volte osservato come la disponibilità di informazioni corrette, esaustive, aggiornate e diffuse in modo capillare sia uno dei fattori chiave per ottenere sviluppo, non solo tecnologico, ma anche sociale. Il continuo aumento della quantità e della complessità delle informazioni da elaborare pone il problema di definire *metodologie corrette ed efficaci per la gestione dell'informazione*: l'accento è quindi posto sulle problematiche legate alla progettazione, realizzazione e validazione delle procedure di elaborazione (il software).

Le competenze della *figura professionale informatico* sono quindi ortogonali sia allo specifico campo di applicazione dell'informazione trattata (economia, ingegneria, chimica farmaceutica, intrattenimento, ecc.), sia ai particolari strumenti che la tecnologia mette a disposizione per l'elaborazione dell'informazione (strumenti che, per se stessi, sono più propriamente oggetto di studi ingegneristici).

Il Corso di Laurea in Informatica è quindi la scelta naturale per tutti coloro i quali, iniziando gli studi universitari, si propongono di acquisire le conoscenze necessarie ad intraprendere un percorso professionale nel quale l'informatica giochi un ruolo di primo piano.

1 Obiettivi formativi

Il Corso di Laurea in Informatica ha come obiettivo la formazione di una figura professionale di esperto nel settore delle Scienze e Tecnologie Informatiche, dotato di competenze tecnico-professionali atte ad un suo rapido inserimento nel mondo del lavoro sia nel settore delle Tecnologie dell'Informazione e della Comunicazione (I.C.T.) che nei diversi settori applicativi basati su tali tecnologie.

I laureati in Informatica sono inoltre preparati per il proseguimento degli studi in corsi di laurea specialistica, di master e di dottorato.

Il corso degli studi intende pertanto fornire, oltre ad una solida preparazione culturale di base, le conoscenze dei metodi, delle tecniche e degli strumenti atti allo sviluppo di sistemi informatici “di qualità”, nonché le competenze necessarie ad un loro proficuo utilizzo nella risoluzione di problemi applicativi specifici.

L’integrazione tra solida preparazione di base e comprensione ed utilizzo delle tecnologie informatiche costituisce elemento fondamentale del corso di studi, indispensabile a produrre quelle competenze necessarie per comprendere l’evoluzione tecnologica, interpretarne i contenuti, individuarne le applicazioni, ampliare e modificare il modo di operare.

Caratteristiche e conoscenze generali

In accordo con gli obiettivi formativi della Classe XXVI, il laureato in Informatica possiede:

- una buona conoscenza di base nei diversi settori dell’Informatica;
- familiarità con il metodo scientifico di indagine e capacità di comprensione degli strumenti matematici di supporto alle competenze informatiche;
- competenze operative e di laboratorio;
- buone capacità applicative;
- capacità di comprensione, integrazione e trasferimento dell’innovazione tecnologica.

Competenze tecnico-professionali

Sulla base di queste caratteristiche e conoscenze generali, e ad integrazione e specializzazione di esse, il corso degli studi in Informatica prevede la creazione di più specifiche competenze tecnico-professionali necessarie per:

- l’analisi e modellizzazione di sistemi complessi;
- l’applicazione delle tecniche e degli strumenti informatici a vari contesti applicativi;
- il progetto e lo sviluppo di sistemi informatici di elevata qualità, anche in relazione ai diversi contesti applicativi;
- la gestione e manutenzione di sistemi informatici;
- il supporto agli utenti nell’utilizzo di strumenti informatici.

2 Sbocchi occupazionali

Grazie al bagaglio di conoscenze ed esperienze acquisite durante il corso degli studi, al momento dell’inserimento nel mondo del lavoro il laureato in Informatica potrà ricoprire una qualunque delle seguenti figure professionali:

- progettista e sviluppatore di sistemi e applicazioni software, in ambiente Internet o di rete locale o per sistemi di calcolo parallelo;
- esperto in tecniche informatiche utilizzabili a fini applicativi;
- gestore di sistemi informatici, in particolare reti;
- istruttore, formatore, divulgatore scientifico.

La possibilità di proseguire gli studi, iscrivendosi a corsi di laurea specialistica, di master e di dottorato, consente anche di intraprendere le carriere della ricerca scientifica e dell'insegnamento, anche a livello universitario.

3 Articolazione del corso di studi

L'ordinamento didattico è formulato con riferimento ai crediti formativi universitari (CFU) secondo la normativa vigente.

La durata del Corso di Laurea in Informatica è di tre anni. Ogni anno di corso è ripartito in due periodi di attività didattica denominati semestri, della durata di almeno 12 settimane, separati da periodi dedicati esclusivamente alla verifica del profitto degli studenti e allo svolgimento delle verifiche in itinere. Il calendario delle lezioni per ogni anno accademico è specificato all'inizio dell'anno nel Manifesto degli Studi.

L'attività dello studente corrisponde al conseguimento di un totale di 180 CFU suddivisi di norma in circa 60 CFU acquisibili per ogni anno di corso.

Le attività formative previste si svolgono sotto forma di lezioni frontali in aula (possibilmente supportate da strumenti audiovisivi), lezioni ed esercitazioni in laboratori attrezzati, progetti individuali coordinati da tutor, tirocini. In particolare, per quanto riguarda le lezioni frontali ogni CFU corrisponde a circa 8 ore di didattica, mentre per le esercitazioni in laboratorio ogni CFU corrisponde a circa 12 ore di didattica.

Organizzazione delle attività formative

L'organizzazione dei corsi per l'anno accademico 2004-2005 è riportata in Tabella 1. Sono previste attività obbligatorie di base, attività a scelta, ed altre attività diverse.

Attività obbligatorie di base.

- Attività (32 CFU) per fornire un'adeguata *formazione matematica di base*, con particolare riferimento all'acquisizione del linguaggio e dei metodi della matematica, ai fondamenti dell'Analisi Matematica, dell'Algebra e della Geometria, e agli elementi di base dell'Analisi Numerica e del Calcolo delle Probabilità;
- attività (6 CFU) per fornire elementi di *fisica*;
- attività (48 CFU) per fornire una buona conoscenza dell'*informatica di base*, con particolare riferimento a Fondamenti e metodologie di Programmazione, Algoritmi e Strutture Dati, Fondamenti dell'Informatica, Architettura degli elaboratori, Sistemi Operativi, Reti di calcolatori, Basi di Dati;

- lezioni ed esercitazioni di *laboratorio* (20 CFU), nonché attività progettuali autonome ed attività di laboratorio individuali, atte ad acquisire la necessaria padronanza dell'utilizzo degli strumenti informatici e a sperimentare nella pratica le nozioni di base acquisite.

Attività a scelta. Sono previste sia attività a scelta guidata che attività a scelta libera.

- Attività a scelta guidata, per un totale di 47 CFU, così articolati:
 - 13 CFU nei settori scientifico-disciplinari INF/01 e ING-INF/05;
 - 34 CFU in un'ampia gamma di settori scientifico-disciplinari, eventualmente indirizzati alla realizzazione di specifici percorsi applicativi. Sono in particolare individuabili un percorso *modellistico-numeric* ed un percorso *chimico-biologico*.
- Attività (9 CFU) scelte liberamente dallo studente nell'ambito degli insegnamenti che costituiscono l'offerta formativa di Ateneo o di altra Istituzione accademica Italiana o straniera.

Altre attività. Sono previste altre attività, per un totale di 21 CFU, volte a creare competenze trasversali, così composte:

- attività (almeno 3 CFU) per fornire conoscenze della lingua inglese;
- attività di formazione (non più di 6 CFU) relative alla cultura di contesto, con particolare riferimento a problematiche di comunicazione, organizzazione delle imprese, aspetti giuridici e professionali dell'informatica;
- attività di tirocinio (non più di 9 CFU) presso aziende, strutture e laboratori, pubblici o privati, in Italia o all'estero, anche nel quadro di apposite convenzioni o accordi internazionali;
- attività (6 CFU) per la prova finale, finalizzate alla preparazione e presentazione di un elaborato scritto (eventualmente in lingua inglese), sotto la guida di un tutor.

L'elenco di tutti i corsi a scelta attivati per l'anno accademico 2004-2005, con indicazione dei CFU e del settore di appartenenza di ogni corso, è riportato sul Manifesto degli studi.

Piani di studio

Entro il 15 ottobre di ogni anno accademico lo studente iscritto al II o III anno di corso dovrà presentare al Consiglio di corso di laurea per l'approvazione il piano di tutte le attività a scelta che intende seguire.

Un piano di studio approvato può essere modificato negli anni accademici successivi.

Propedeuticità

Le eventuali propedeuticità tra insegnamenti sono indicate nel Manifesto degli studi nell'elenco dei corsi attivati.

TABELLA 1: ORGANIZZAZIONE DEI CORSI

Periodo	Corso	Area	CFU
I Anno I semestre	Algebra	MAT	6
	Analisi Matematica 1	MAT	7
	Architettura degli Elaboratori	INF	6
	Fondamenti di Programmazione	INF	6
	Laboratorio di Programmazione	INF	4
I Anno II Semestre	Algebra Lineare e Geometria	MAT	7
	Algoritmi e Strutture Dati	INF	6
	Lab. di Algoritmi e Strutture Dati	INF	4
	Fisica 1	FIS	6
	Inglese		3
Totale CFU I Anno:			55
II Anno I semestre	Calcolo Probabilità e Statistica	MAT	6
	Fondamenti dell'Informatica	INF	6
	Metodologie di Programmazione	INF	6
	Corsi a scelta guidata		9
	Corsi a scelta libera		3
II Anno II Semestre	Calcolo Numerico 1	MAT	6
	Lab. Computaz. Numerico	MAT	2
	Sistemi Operativi	INF	6
	Lab. di Sistemi Operativi	INF	4
	Corsi a scelta guidata		12
Totale CFU II Anno:			60
III Anno I semestre	Reti di Calcolatori	INF	6
	Lab. di Programmazione di Rete	INF	4
	Corsi a scelta guidata		14
	Corsi a scelta libera		6
III Anno II Semestre	Basi di Dati	INF	6
	Lab. di Basi di Dati	INF	2
	Corsi a scelta guidata		12
	Tirocinio e altre abilità		9
	Prova finale		6
Totale CFU III Anno:			65

Tirocinio

Il *Tirocinio* prevede lo svolgimento di un'attività di lavoro individuale, a prevalente carattere pratico e/o sperimentale, da effettuarsi presso Aziende o Enti *esterni* o presso Laboratori di ricerca *interni* all'Ateneo. L'attività di tirocinio può essere sostituita, anche parzialmente, da un secondo corso di lingua straniera (3 CFU) e/o da uno o più corsi di "cultura di contesto" (per un massimo di 6 CFU). Gli esami relativi al secondo corso di lingua straniera e ai corsi di "cultura di contesto" possono essere sostenuti in qualsiasi anno del corso degli studi.

4 Frequenza

La frequenza ai corsi è un diritto/dovere dello studente.

La frequenza dei corsi di Laboratorio fondamentali è obbligatoria. Per essere ammessi all'esame di tali corsi è necessario averne frequentato almeno il 75% delle ore di lezione ed esercitazioni. Per studenti lavoratori, portatori di handicap e rappresentanti negli organi collegiali dell'Università, potranno essere concordate modalità diverse di assolvimento degli obblighi di frequenza.

Per tutti gli altri corsi lo studente ottiene automaticamente la firma di frequenza al termine del periodo nel quale l'insegnamento previsto nel piano degli studi è stato impartito.

Eventuali deroghe alle regole concernenti la frequenza dei corsi potranno essere stabilite all'inizio di ciascun anno accademico con pubblicazione sul Manifesto degli studi.

Studenti lavoratori

Gli studenti lavoratori che per cause motivate non possono seguire le lezioni possono chiedere la dispensa, anche parziale, dall'obbligo di frequenza dei corsi che lo prevedono. Resta fermo l'obbligo di concordare con i docenti dei corsi interessati modalità alternative di acquisizione delle conoscenze e capacità derivanti dalla frequenza dei suddetti corsi.

I docenti del corso di laurea si impegnano a seguire con particolare cura il percorso formativo degli studenti lavoratori fornendo il materiale didattico necessario e rendendosi disponibili per colloqui anche al di fuori del normale orario di ricevimento.

5 Verifica dell'apprendimento ed esami di profitto

I CFU corrispondenti a ciascuna attività formativa sono acquisiti dallo studente previo superamento dell'esame di profitto.

Ognuna delle seguenti coppie di insegnamenti prevede un esame di profitto unico:

- *Algoritmi e Strutture Dati* e *Laboratorio di Algoritmi e Strutture Dati*
- *Basi di Dati* e *Laboratorio di Basi di Dati*

- *Calcolo Numerico 1 e Laboratorio Computazionale Numerico*
- *Fondamenti di Programmazione e Laboratorio di Programmazione*
- *Reti di Calcolatori e Laboratorio di Programmazione di Rete*
- *Sistemi Operativi e Laboratorio di Sistemi Operativi*

Gli esami di profitto possono avere forma orale o scritta o mista ovvero possono consistere in prove pratiche o di laboratorio o progetti svolti dagli studenti. L'esame di profitto può essere la conclusione di un processo valutativo sviluppato durante il corso di insegnamento con prove in itinere, colloqui ed esercitazioni.

La valutazione per gli esami di profitto è espressa di norma con una votazione in trentesimi, con eventuale lode. Gli esami dei corsi relativi alla “cultura di contesto”, nonché quelli dei corsi di lingua straniera, e l'attività di tirocinio, prevedono una valutazione espressa in termini di un giudizio globale di idoneità, ottenuto mediante prove in itinere e/o progetti e/o esame finale.

Gli esami di profitto finali per ciascun insegnamento si svolgono nei periodi di interruzione delle attività didattiche e comunque a partire dalla fine del semestre nel quale il corso è stato svolto e nelle due sessioni successive dello stesso anno accademico. In particolare: le sessioni di esame per i corsi svolti nel I semestre sono di norma tenute nei mesi di Gennaio—Febbraio (I sessione), Giugno—Luglio (II sessione) e Settembre—Ottobre (III sessione); le sessioni di esame per i corsi svolti nel II semestre sono di norma tenute nei mesi di Giugno—Luglio (I sessione), Settembre—Ottobre (II sessione) e Gennaio—Febbraio (III sessione). Ciascuna sessione d'esame ha la durata di almeno quattro settimane.

6 Prova finale per il conseguimento del titolo

La prova finale consiste nella stesura e nella presentazione di un elaborato scritto, preparato sotto la guida di un relatore. Tale elaborato riporta un lavoro individuale svolto all'interno dell'Università o presso un'Azienda o un Ente esterno in un periodo di tirocinio, o di stage, previa approvazione del Consiglio di corso di laurea.

L'argomento dell'elaborato è assegnato su richiesta dello studente da un docente dell'Ateneo, che funge da relatore, almeno due mesi prima della data prevista per la sessione di laurea.

L'esame di laurea consiste nella discussione di fronte ad una commissione nominata dal Consiglio di corso di laurea dell'elaborato scritto preparato. La valutazione del candidato da parte della commissione di laurea, espressa in centodecimi con eventuale lode, avviene tenendo conto delle risultanze dell'intera carriera universitaria, comprensiva della prova finale, e determina il possibile incremento del voto di Laurea finale rispetto alla media pesata (massimo 11 punti). Nel calcolo della media si assumono come pesi i CFU assegnati a ciascuna attività per la quale è prevista una votazione in trentesimi.

Per ciascun anno accademico sono previste tre sessioni ordinarie di esami di laurea: estiva, autunnale e invernale. La prima sessione utile per sostenere l'esame di laurea è di norma quella al termine del secondo periodo del terzo anno di corso. La sessione invernale è una sessione dell'anno accademico precedente

a quello in cui si svolge e termina alla fine di marzo. Il Consiglio può istituire sessioni straordinarie di Laurea.

Per essere ammesso all'esame di laurea lo studente deve aver acquisito almeno 174 CFU e presentato la relazione scritta, controfirmata dal Relatore, almeno 10 giorni prima della data fissata per la discussione.

7 Modalità di accesso al corso di laurea

Immatricolazioni e valutazione iniziale

Il titolo di ammissione al Corso di Laurea è quello previsto dalla legge.

Per quanto riguarda le modalità e le date per la presentazione delle domande di immatricolazione, per l'eventuale prova di ingresso, per le attività formative propedeutiche (precorsi) e per le altre eventuali attività di sostegno si rimanda al Manifesto degli studi.

Inserimenti e trasferimenti

Gli studenti iscritti ad altro corso di laurea della stessa o di altre Università e gli studenti già in possesso di altro titolo accademico che intendano iscriversi al corso di Laurea in Informatica di Parma dovranno presentare opportuna domanda entro i termini e con le modalità fissate dal Manifesto degli studi.

Completata l'iscrizione, tali studenti potranno chiedere il riconoscimento degli esami già sostenuti. Le domande verranno esaminate dal Consiglio di corso di laurea, previa presentazione di copia dei programmi dei corsi per i quali si richiede la convalida di frequenza ed esami e di autocertificazione attestante la veridicità degli stessi. Il Consiglio di corso valuterà caso per caso la carriera di studio fino a quel momento seguita dallo studente al fine di un eventuale riconoscimento di crediti acquisiti, indicando nel contempo l'anno di corso al quale lo studente viene iscritto e l'eventuale debito formativo da colmare.

L'iscrizione ad un anno successivo al primo può avvenire soltanto nel caso in cui risultino convalidati almeno il 30% (in termini di CFU) degli esami degli anni precedenti.

I crediti acquisiti in insegnamenti o attività che non trovano un corrispettivo nel piano di studi del presente corso di laurea potranno essere riconosciuti, a domanda, tra quelli a scelta dello studente fino ad un massimo di 9 CFU.

Il voto con cui vengono riconosciuti i crediti corrisponde al voto conseguito nell'esame che viene convalidato, ovvero alla media aritmetica dei voti nel caso di convalida di più esami (moduli) in un unico esame.

Iscrizione agli anni successivi al primo

Per il passaggio all'anno di corso successivo (degli studenti già iscritti al Corso di Laurea in Informatica di questa università) non sono previsti vincoli, se non l'avvenuto pagamento delle tasse di iscrizione.

Lo studente che non consegue il titolo al termine del terzo anno di corso viene iscritto come fuori corso.

8 Corsi attivati nell'A.A. 2004-2005

Nell'anno accademico 2004-2005 sono attivati i corsi obbligatori elencati in Tabella 2 e i corsi a scelta guidata elencati in Tabella 3.

TABELLA 2: CORSI OBBLIGATORI ATTIVATI

<i>N.</i>	<i>Titolo</i>	<i>CFU</i>
1	Algebra	6
2	Algebra Lineare e Geometria	7
3	Algoritmi e Strutture Dati 1	6
4	Analisi Matematica 1	7
5	Architettura degli Elaboratori	6
6	Basi di Dati	6
7	Calcolo Numerico 1	6
8	Calcolo Probabilità e Statistica	6
9	Fisica 1	6
10	Fondamenti dell'Informatica	6
11	Fondamenti di Programmazione	6
12	Inglese	3
13	Laboratorio Computazionale Numerico	2
14	Laboratorio di Algoritmi e Strutture Dati	4
15	Laboratorio di Basi di Dati	2
16	Laboratorio di Programmazione	4
17	Laboratorio di Programmazione di Rete	4
18	Laboratorio di Sistemi Operativi	4
19	Metodologie di Programmazione	6
20	Reti di Calcolatori	6
21	Sistemi Operativi	6

Nelle pagine seguenti sono indicate, così come comunicate dai rispettivi docenti, le informazioni riguardanti la finalità, il programma di massima ed i testi consigliati per tutti i corsi attivati nell'anno accademico 2004-2005. Ulteriori informazioni (ad esempio, l'orario settimanale e la sede delle lezioni per i vari corsi) verranno pubblicati, non appena disponibili, sul sito web del corso di laurea, all'indirizzo

<http://www.cs.unipr.it/Informatica>

TABELLA 3: CORSI A SCELTA ATTIVATI

<i>N.</i>	<i>Titolo</i>	<i>CFU</i>
22	Algoritmi e Strutture Dati 2	4
23	Analisi e Verifica del Software	6
24	Analisi Matematica 2	6
25	Bioinformatica	5
26	Biologia per Bioinformatica	5
27	Calcolo Numerico 2	6
28	Calcolo Parallelo	4
29	Chimica	6
30	Complementi di Analisi Matematica	6
31	Elettronica 1	4
32	Elettronica 2	4
33	Fisica 2	6
34	Geometria	6
35	Grafica Computazionale Tecnica	5
36	Informatica in Azienda	3
37	Informatica Teorica	5
38	Ingegneria del Software	6
39	Intelligenza Artificiale	6
40	Laboratorio di Geoinformatica	5
41	Lambda Calcolo e Combinatori	4
42	Linguaggi Dichiarativi	4
43	Logica Matematica	6
44	Modellazione e Simulazioni Numeriche	4
45	Modelli e Sistemi Dinamici	6
46	Modellistica Molecolare	4
47	Reti Logiche	5
48	Ricerca Operativa	6
49	Scrittura Matematica e Informatica	3
50	Secondo corso di lingua straniera	3
51	Sistemi Informativi Geografici	3
52	Strumenti per Applicazioni Web	5
53	Teoria dei Numeri e Crittografia	4
54	Teoria dei Segnali	5
55	Teoria dell'Informazione	6

ALGEBRA

Docente: G. Ferrero

Crediti (CFU): 6

Programma del corso:

- *Introduzione.*
Origine e momenti chiave dello sviluppo dell'Algebra.
Idea di operazione.
Strutture con due operazioni, operazioni esterne, campi e spazi vettoriali.
Gruppidi, sottogruppidi, prodotti diretti.
Gruppidi con identità od elementi particolari.
Omomorfismi, congruenze, gruppidi quoziente.
Teoremi fondamentali sugli omomorfismi.
Elementi di combinatoria.
- *Strutture classiche elementari.*
Interi relativi, numeri razionali.
Divisione euclidea.
Interi modulo n , prova del nove.
Gruppi ciclici.
Laterali e sottogruppi normali di un gruppo. Teorema di Lagrange.
Piccolo teorema di Pascal. Applicazioni.
Teorema di fattorizzazione di Gauss.
Teoremi di Euclide sui numeri primi.
Polinomi e funzioni polinomiali.
Aritmetica dei polinomi. Teorema di Ruffini.
Equazioni di secondo grado.
- *Anelli.*
Sottoanelli e ideali. Anelli quozienti.
Ideali: principali, massimali, primi e primari.
Anelli a ideali principali.
Divisibilità e fattorizzazione.
Equazioni polinomiali in un campo. Polinomi riducibili e irriducibili.
Estensioni algebriche e trascendenti. Come funziona la razionalizzazione.
Campi finiti.

Testi consigliati:

1. L. Childs. *Algebra*, ETS editrice, 1983.
2. C. Cotti, G. Ferrero. *Strutture matematiche elementari*, RSAS, Parma, 1994.
3. S. Franciosi, F. de Giovanni, *Elementi di Algebra*, Aracne, 1992.
4. J. D. Lipson. *Elements of algebra and algebraic computing*, Addison-Wesley.
5. Van Der Waerden. *Algebra*, (Varie edizioni, inglesi o tedesche).

ALGEBRA LINEARE E GEOMETRIA

Docente: L. Bertani

Crediti (CFU): 7

Programma del corso:

- Spazi vettoriali su un campo, sottospazi, indipendenza lineare, basi, cambiamento di base.
- Matrici, determinante e rango, sistemi lineari.
- Omomorfismi di spazi vettoriali e matrici; nucleo e immagine. Forme lineari.
- Autovalori, autovettori e diagonalizzazione di una matrice.
- Prodotti scalari, spazi vettoriali euclidei.
- Rette, distanze, angoli, parallelismo e ortogonalità in \mathbb{R}^2 ; rette, piani, distanze, angoli parallelismo e ortogonalità in \mathbb{R}^3 .

Testi consigliati:

1. M. Abate. *Algebra Lineare*, McGraw-Hill.
2. E. Bargerò Rivelli. *Esercizi di geometria e Algebra Lineare*, Levrotto & Bella, Torino, 1983;
3. G. Accascina - V. Villani. *Esercizi di Algebra Lineare*, ETS.

ALGORITMI E STRUTTURE DATI 1

(+ LABORATORIO)

Docente: G. Lotti

Docente per il laboratorio: G. Lotti

Crediti (CFU): 6 per la parte teorica, 4 per il laboratorio

Programma del corso:

- *Analisi di algoritmi e complessità.*
Dimensione dei dati di un problema. Ordini di grandezza delle funzioni. Caso pessimo e medio. Limiti superiori ed inferiori alla complessità di un problema. Tecniche per la dimostrazione di limiti inferiori. Complessità polinomiale e superpolinomiale. Relazioni di ricorrenza: metodi di soluzione e teorema fondamentale.
- *Modelli di calcolo sequenziale.*
Macchina ad accesso casuale (RAM). Risorse in spazio e tempo. Criteri di costo uniforme e logaritmico. Altri modelli di calcolo.
- *Strutture dati elementari.*
Strutture elementari: liste, pile, code, heap e relative operazioni fondamentali. Esecuzione iterativa delle chiamate ricorsive: record di attivazione delle chiamate, loro gestione mediante una pila e analisi dello spazio di memoria utilizzato. Algoritmi e strutture dati per la gestione e manipolazione di insiemi: tabelle hash, alberi binari di ricerca, bilanciamento, skip-lists e B-alberi. Algoritmi e strutture dati per il problema Union-Find. Code con priorità, heap.
- *Progetto di algoritmi.*
Tecniche di progettazione di algoritmi ed esempi di applicazione: tecnica divide et impera, backtrack, greedy, programmazione dinamica. Algoritmo di Karatzuba-Hoffman per il prodotto di interi. Prodotto di una sequenza di matrici. Codici di Huffman.
- *Algoritmi di ricerca e ordinamento.*
Generalità sul problema dell'ordinamento. Ordinamento interno per confronti: numero minimo di confronti necessari per ordinare n elementi. Algoritmi primitivi di ordinamento: selection-sort, insertion-sort, bubble-sort. L'algoritmo heapsort. Algoritmi ricorsivi: mergesort, quicksort. Analisi del quicksort nel caso medio. Implementazione iterativa di quicksort e ottimizzazione dello spazio di memoria. Algoritmi lineari non basati sul confronto: counting-sort, radix-sort, bucket-sort. Determinazione dell'elemento medio.
- *Algoritmi elementari sui grafi.*
Tecniche di rappresentazione di grafi orientati e non orientati. Algoritmi di visita in ampiezza e profondità, alberi di copertura. Algoritmi di visita su alberi. Calcolo delle componenti fortemente connesse. Cammini minimi

su grafi. Algoritmi per la determinazione di ordinamenti topologici, alberi di copertura minimi, cammino minimo da una sorgente, cammini minimi da sorgenti multiple.

Testi consigliati:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest. *Introduzione agli algoritmi*, Vol. 1, 2, 3, Jackson, 1994.
2. A. Bertossi. *Algoritmi e strutture dati*, UTET, 2000.
3. G. Fiorentino, M. Laganà, F. Romani, F. Turini. *C e Java: laboratorio di programmazione*, McGraw-Hill, 1997.
4. R. Sedgwick. *Algorithms in C++* (terza edizione), Addison-Wesley, 2003.

ALGORITMI E STRUTTURE DATI 2

Docente: G. Lotti

Crediti (CFU): 4

Programma del corso:

- *Algoritmi greedy*: activity selection, zaino frazionario, scheduling di job.
- *Programmazione dinamica*: zaino intero, parentesizzazione ottima, sotto-sequenza comune di lunghezza massima.
- *Algoritmi algebrici*: algoritmo di Strassen. Polinomi, interi ed FFT.
- *String matching esatto e approssimato*: algoritmi di Knuth-Morris e Pratt, Rabin e Karp, Boyer e Moore. Suffix trees e applicazioni, algoritmo di Sellers, algoritmo di Landau e Vishkin, algoritmo di Chang e Lawler.
- *Algoritmi di teoria dei numeri (cenni)*: crittografia a chiave pubblica. RSA.
- *Problemi computazionalmente difficili*: la classe P, la classe NP, riducibilità polinomiale, problemi NP-completi, il teorema di Cook. Algoritmi di approssimazione.
- *Algoritmi probabilistici*: test di primalità di Rabin, moltiplicazione di matrici.

Testi consigliati:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest. *Introduction to Algorithms 2nd edition*, MIT Press, 2001.
2. A. Bertossi. *Algoritmi e strutture dati*, UTET, 2000.
3. A. Bernasconi, B. Codenotti. *Introduzione alla complessità computazionale*, Springer Verlag, 1998.
4. P. Ferragina, F. Luccio. *Crittografia: Principi, Algoritmi, Applicazioni*, Bollati Boringhieri, 2001.
5. D. Gusfield. *Algorithms on Strings, Trees and Sequences*, Cambridge University Press, 1997.
6. V. V. Vazirani. *Approximation Algorithms*, Springer Verlag, 2001.
7. R. Motwani, P. Raghavan. *Randomized algorithms*, Cambridge University Press, 1995.
8. Appunti delle lezioni a cura del docente.

ANALISI E VERIFICA DEL SOFTWARE

Docente: R. Bagnara

Collaboratore didattico: E. Zaffanella

Crediti (CFU): 6

Finalità: la crescente dipendenza della società dalle applicazioni informatiche fa sì che l'analisi e la verifica della correttezza dei sistemi complessi rappresenti sempre di più un fattore critico del processo di sviluppo. Il malfunzionamento dei sistemi, siano essi hardware, software o protocolli di comunicazione, può comportare danni rilevanti di ogni genere: dalla perdita finanziaria alla perdita di vite umane. Inoltre, quando i difetti non sono rilevati prima dell'impiego del sistema, l'applicazione di eventuali misure correttive è, quando possibile, ben più difficile e costosa. Esempi dal recente passato includono il *millennium bug*, gli errori di alcune versioni del processore *Pentium*, lo scoperto da 32 miliardi di dollari alla *N.Y. Bank*, il fallimento iniziale del vettore *Ariane 5*, e gli incidenti mortali del *Therac-25*.

Il corso intende fornire una prima introduzione alle tecniche che stanno alla base dell'analisi automatica del software e della verifica formale assistita dal calcolatore.

Programma del corso:

- Introduzione all'analisi e alla verifica del software.
- Specifiche e proprietà di programmi.
- Logica di Hoare e verifica di programmi sequenziali.
- Semantica operativa strutturata.
- Interpretazione astratta.
- Analisi statica di programmi.

ANALISI MATEMATICA 1

Docente: L. Lorenzi

Collaboratore didattico: A. Lunardi

Crediti (CFU): 7

Programma del corso:

- *I numeri reali.*
Definizione assiomatica dei numeri reali, massimo, minimo, estremo superiore e inferiore; parte intera e modulo dei numeri reali; potenze, radici, radici n-esime dei numeri non negativi; numeri razionali e irrazionali; intervalli, distanza; intorni, punti di accumulazione, punti isolati, punti interni; insiemi chiusi, insiemi aperti, frontiera.
- *Funzioni.*
Funzioni iniettive, surgettive, biunivoche, funzione inversa; grafici; funzioni reali di variabile reale, funzioni monotone, funzioni esponenziali e logaritmiche; funzioni trigonometriche.
- *Limiti.*
Limiti di funzioni con valori reali, unicità del limite, limiti delle restrizioni; limite della somma, prodotto, quoziente di due funzioni; permanenza del segno, teoremi di confronto; limite destro e sinistro; limiti delle funzioni monotone; ordini di infinitesimi e di infiniti, asintoti.
- *Funzioni continue.*
Continuità di funzioni reali di variabile reale, restrizioni di funzioni continue, composizione di funzioni continue; somma, prodotto, quoziente di funzioni continue; esempi di funzioni continue; discontinuità, esempi di funzioni discontinue; teorema degli zeri; continuità e intervalli; continuità e monotonia; continuità delle funzioni inverse; teorema di Weierstrass.
- *Calcolo differenziale.*
Rapporti incrementali, derivate, derivate destre e sinistre; significato geometrico della derivata; regole di derivazione: derivate della somma, prodotto, quoziente di due funzioni; derivate di funzioni composte e di funzioni inverse; derivate delle funzioni elementari; massimi e minimi relativi; punti stazionari; relazione fra monotonia e segno della derivata; teoremi di Rolle, Lagrange e loro interpretazione geometrica, teoremi di Cauchy e di De l'Hôpital; funzioni convesse, derivate delle funzioni convesse, relazione fra convessità e segno della derivata seconda; formula di Taylor con resto di Peano, di Lagrange e in forma integrale; studio dei massimi e minimi locali col calcolo delle derivate successive.
- *Integrali.*
Partizioni di un intervallo; integrale superiore e inferiore, funzioni integrabili in un intervallo, integrabilità di funzioni continue e di funzioni

monotone; interpretazione geometrica dell'integrale; proprietà degli integrali; media di una funzione integrabile; integrali su intervalli orientati; teorema fondamentale del calcolo integrale; primitive, integrali indefiniti; integrazione per parti e per sostituzione; integrali di funzioni razionali.

Testi consigliati:

1. J. Cecconi, G. Stampacchia. *Analisi Matematica 1*, Liguori, 1974;
2. M. Giaquinta, G. Modica. *Analisi Matematica. 1: Funzioni di una variabile*, Pitagora, 1998;
3. E. Giusti. *Analisi Matematica 1*, Boringhieri, 1983.

ANALISI MATEMATICA 2

Docente: A. Lunardi

Collaboratore didattico: L. Lorenzi

Crediti (CFU): 6

Programma del corso:

- *Numeri complessi.*
Definizione, operazioni elementari e loro rappresentazione grafica.
- *Successioni.*
Principio di induzione; successioni di numeri reali e complessi, successioni convergenti, unicità del limite; sottosuccessioni; successioni di Cauchy; successioni infinitesime, successioni divergenti; somme, prodotti, quozienti, permanenza del segno, teoremi di confronto; successioni monotone; il numero e ; il numero π , successioni definite per ricorrenza; massimo e minimo limite. Numeri razionali e irrazionali; rappresentazione decimale; non numerabilità dei reali, densità dei razionali nei reali. Teorema di Bolzano-Weierstrass e compattezza in \mathbb{R} . Potenze con esponente reale.
- *Serie.*
Serie convergenti, divergenti, indeterminate; criterio di Cauchy per le serie; criterio di confronto, del rapporto, della radice; serie assolutamente convergenti, riordinamenti; serie a termini di segno alterno, criterio di Leibniz; esempi: serie geometriche, serie telescopiche, serie armonica, serie armonica generalizzata e serie armonica a segni alterni, serie esponenziali.
- *Complementi di calcolo.*
Integrali generalizzati di funzioni illimitate e su intervalli illimitati; criterio di Cauchy e criterio di confronto; criterio integrale di convergenza per serie a termini positivi. Funzioni uniformemente continue.

Testi consigliati:

1. J. Cecconi, G. Stampacchia. *Analisi Matematica 1*, Liguori, 1974;
2. M. Giaquinta, G. Modica. *Analisi Matematica. 2: Approssimazione e processi discreti*, Pitagora, 1998;
3. E. Giusti. *Analisi Matematica 1*, Boringhieri, 1983.

ARCHITETTURA DEGLI ELABORATORI

Docente: R. Covati

Crediti (CFU): 6

Programma del corso:

- Evoluzione storica degli elaboratori. Architettura di un elaboratore, macchina di Von Neumann. Esempio di linguaggio macchina per un calcolatore di Von Neumann.
- Sistemi di numerazione binario, esadecimale e ottale. Rappresentazione binaria dei numeri reali, standard IEEE. Codici di caratteri.
- Porte e circuiti logici. Cenni di algebra booleana e di minimizzazione di funzioni logiche. Reti combinatorie e sequenziali. Il clock. Esempi di circuiti logici: contatori, multiplexer, decodificatori, comparatori, shifter, addizionatori, ALU; latch e flip flop, registri e memorie. I circuiti integrati.
- Architettura di un calcolatore moderno. Memorie statiche e dinamiche, RAM, ROM, EPROM, etc. I bus dei calcolatori, sincroni e asincroni, arbitraggio del bus. Architetture di bus: ISA, PCI, AGP, etc. Gestione degli interrupt. Microprocessori Risc e Cisc. Panoramica dei moderni microprocessori. Gestione dell'input/output: seriale, parallelo, SCSI, USB, Firewire. DMA. Memorie di massa.
- Prestazioni di un sistema. Legge di Amdahl. Miglioramento delle prestazioni: la pipeline, la memoria cache. Cenni di architetture parallele.
- Accenni di Microprogrammazione. Catena di compilazione.
- Introduzione ad un sistema operativo. Processi, memoria virtuale.
- Elementi di linguaggio assembler x86. Registri utilizzati. Istruzioni di spostamento di dati, istruzioni aritmetiche, booleane, rotazione e traslazione, test, trasferimento di controllo, etc. Diversi tipi di indirizzamento della memoria. Rudimenti di utilizzo del sistema operativo Linux: assembler e compilatori. Set di istruzioni aggiuntive nei moderni microprocessori: MMX, SSE, SSE2, 3Dnow. Programmazione C usando direttive di inline assembler.

Testi consigliati:

1. Andrew S. Tanenbaum. *Architettura del Computer, un approccio strutturato*, UTET Libreria, 2000.
2. Giacomo Bucci. *Architetture dei Calcolatori Elettronici*, McGraw-Hill, 2001.
3. John P. Hayes. *Computer Architecture and Organization*, McGraw-Hill.

BASI DI DATI (+ LABORATORIO)

Docente: E. Zaffanella

Docente per il laboratorio: E. Zaffanella

Crediti (CFU): 6 per la parte teorica, 2 per il laboratorio

Finalità: nella prima parte del corso verrà fornita un'introduzione all'utilizzo dei sistemi di gestione di basi di dati, con particolare riferimento ai sistemi che adottano il modello relazionale. Argomento della seconda parte del corso saranno le metodologie e tecniche di progettazione di una base di dati relazionale. La parte di *Laboratorio*, che prevede l'introduzione ad uno specifico sistema di gestione di basi di dati, consisterà in esercitazioni pratiche e nella preparazione di un progetto da discutere in sede di esame.

Programma del corso:

- Introduzione ai sistemi di gestione di basi di dati.
- Il modello relazionale dei dati.
- Algebra e calcolo relazionale.
- Il linguaggio SQL.
- Utilizzo di SQL nei linguaggi di programmazione.
- Gestione delle transazioni.
- Sicurezza e gestione dei diritti di accesso.
- Basi di dati attive.
- Metodologie per il progetto di basi di dati: progettazione concettuale, logica e fisica.
- Il modello Entità-Relazione.
- Normalizzazione di schemi di basi di dati.

Testi consigliati:

1. P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone. *Basi di dati: modelli e linguaggi di interrogazione*, McGraw-Hill, 2002.

BIOINFORMATICA

Docente: R. Percudani

Crediti (CFU): 5

Programma del corso:

- Evoluzione di DNA e proteine.
- Metodi di confronto di sequenze biologiche: allineamento a coppie.
- Ricerca in banca dati di sequenze biologiche.
- Allineamento multiplo.
- Predizione delle caratteristiche biochimiche delle proteine.
- Motivi e caratteristiche delle sequenze di DNA.
- Predizione filogenetica.
- Predizione della struttura di RNA e Proteine.

Testi consigliati:

1. G. Valle et al. *Introduzione alla bioinformatica*, Zanichelli, 2003.
2. A. Tramontano. *Bioinformatica*, Zanichelli 2002.
3. D. W. Mount. *Bioinformatics: Sequence and Genome analysis*, CSHL Press, 2001.
4. L. Patty. *Protein Evolution*, Blackwell Science, 1999.

BIOLOGIA PER BIOINFORMATICA

Docente: G. Dieci

Crediti (CFU): 5

Programma del corso:

- LE PROTEINE

- I componenti molecolari della materia vivente. Le macromolecole: aspetti chimici e biologici. Acidi nucleici e proteine come depositari della diversità biologica.
- I venti amminoacidi presenti nelle proteine. Il legame peptidico.
- I livelli di struttura delle proteine: struttura primaria, secondaria, terziaria e quaternaria.
- Denaturazione e ripiegamento (*folding*) delle catene polipeptidiche. Modificazioni covalenti delle proteine: fosforilazione, glicosilazione, proteolisi.
- Relazioni fra la struttura delle proteine e la loro funzione. Gli enzimi e la catalisi enzimatica. Interazioni complementari tra proteine e ligandi: le immunoglobuline. Cambiamenti conformazionali e regolazione della funzione proteica.
- Introduzione ai principali metodi di separazione, purificazione e analisi delle proteine.

- LA CELLULA

- Le membrane cellulari e le proteine di membrana.
- Cellule procariotiche e cellule eucariotiche. Organizzazione strutturale e funzionale della cellula eucariotica: gli organelli.
- Il nucleo: conservazione e trasmissione delle informazioni genetiche.
- Ribosomi, reticolo endoplasmico, apparato di Golgi: biosintesi e smistamento delle proteine.
- Mitochondri e cloroplasti: trasformazioni energetiche.
- Cromosomi, ciclo cellulare, divisione cellulare.

- ACIDI NUCLEICI, INFORMAZIONE ED EREDITARIETÀ

- Struttura e nomenclatura delle basi azotate e dei nucleotidi. Lo scheletro covalente degli acidi nucleici.
- La doppia elica del DNA.
- Denaturazione, rinaturazione e ibridazione del DNA. Analisi degli acidi nucleici mediante elettroforesi.
- La replicazione del DNA: caratteristiche fondamentali e basi chimiche. Le DNA polimerasi e le altre proteine replicative. Meccanismi di replicazione.

- Il sequenziamento del DNA. La reazione polimerasica a catena (PCR). Ricombinazione e riparazione del DNA (cenni).
- L'RNA: struttura e funzioni. La sintesi DNA-dipendente di RNA (trascrizione). RNA polimerasi e promotori. I fattori di trascrizione. Modificazioni dell'RNA dopo la trascrizione.
- Il codice genetico. La sintesi proteica.
- I livelli di regolazione dell'espressione genica.
- Enzimi di restrizione e di modificazione del DNA e loro impiego nella tecnologia del DNA ricombinante. Sintesi RNA-dipendente di DNA (trascrittasi inversa).
- Geni e genomi. Organizzazione del DNA nei virus, nei batteri e negli eucarioti.
- Evoluzione molecolare.

Testi consigliati:

1. Purves, Sadawa, Orians, Heller. *Biologia: la cellula*, Zanichelli, Bologna, 2001.
2. Purves, Sadawa, Orians, Heller. *Biologia: l'informazione e l'ereditarietà*, Zanichelli, Bologna, 2001.

CALCOLO NUMERICO 1

(+ LABORATORIO COMPUTAZIONALE NUMERICO)

Docente: A. Aimi

Docente per il laboratorio: docenza da assegnare

Crediti (CFU): 6 per la parte teorica, 2 per il laboratorio

Programma del corso:

- *Stabilità, condizionamento e analisi dell'errore.*
Buona posizione e numero di condizionamento di un problema. Stabilità di metodi numerici. Relazioni tra stabilità e convergenza. Analisi a priori ed a posteriori. Sorgenti di errori nei modelli computazionali. Rappresentazione dei numeri. Il sistema posizionale e il sistema dei numeri floating-point. Arrotondamento di un numero reale nella sua rappresentazione macchina. Operazioni di macchina effettuate in virgola mobile.
- *Interpolazione polinomiale di funzioni e dati.*
Il problema dell'interpolazione polinomiale. Forma di Lagrange e di Newton del polinomio interpolatore. Interpolazione lineare iterata. L'errore di interpolazione. Limiti dell'interpolazione polinomiale su nodi equidistanti e controesempio di Runge. Stabilità dell'interpolazione polinomiale. Interpolazione di Hermite. Spline lineari interpolatorie. Convergenza.
- *Integrazione numerica.*
Formule di quadratura interpolatorie. Formule di Newton-Cotes semplici e composite. Stime dell'errore. Integrazione automatica.
- *Algebra lineare Numerica 1.*
Analisi di stabilità per sistemi lineari. Il numero di condizionamento di una matrice. Risoluzione di sistemi triangolari. Il metodo di eliminazione gaussiana. L'effetto degli errori di arrotondamento. Pivoting. Fattorizzazione LU. Matrici simmetriche e definite positive: fattorizzazione di Cholesky. Calcolo dell'inversa di una matrice. Matrici tridiagonali. Sistemi tridiagonali a blocchi. Scaling.
- *Ricerca di radici di equazioni non lineari.*
Condizionamento di una equazione non lineare. Il metodo di bisezione. I metodi delle corde, secanti, Regula Falsi. Teoremi di convergenza. Criteri di arresto. Il metodo Newton. Convergenza locale. Radici di polinomi algebrici. Il metodo della successione di Sturm. Il metodo di Newton-Horner. Il metodo di Bairstow. Il metodo di Newton per sistemi.

Programma per la parte di Laboratorio:

- *Introduzione a Matlab 6.*
Matlab come ambiente di calcolo. Array e Matrici. Files, funzioni e strutture dati. Matlab come linguaggio di programmazione. Diagrammi. Grafica. Funzioni Matlab per la risoluzione di problemi del Calcolo Numerico.

- Matlab come strumento per l'implementazione e l'analisi di algoritmi del Calcolo Numerico.

Testi consigliati:

1. V. Comincioli. *Analisi numerica*, Springer.
2. G. Naldi, L. Pareschi, G. Russo. *Introduzione al Calcolo Scientifico. Metodi ed applicazioni con Matlab*, McGraw-Hill.
3. A. Quarteroni, R. Sacco, F. Saleri. *Matematica numerica*, Springer.
4. J. Stoer. *Introduzione all'analisi numerica*, Vol. I, Zanichelli.
5. W. J. Palm. *Matlab 6 per l'ingegneria e le scienze*, McGraw-Hill, 2001.

CALCOLO NUMERICO 2

Docente: A. Aimi

Crediti (CFU): 6

Programma del corso:

- *Approssimazione di funzioni e dati.*
Spline lineari e cubiche interpolatorie. Teorema di convergenza. Splines cardinali e B-Splines. Spline parametriche. Interpolazione trigonometrica. Polinomi ortogonali e approssimazione di una funzione nel senso dei minimi quadrati. I minimi quadrati discreti.
- *Integrazione numerica.*
Integrazione gaussiana su intervalli limitati e intervalli illimitati. Stime dell'errore. Integrazione in più dimensioni.
- *Algebra Lineare Numerica 2.*
Sistemi sovradeterminati: la fattorizzazione QR. Costruzione di metodi iterativi lineari. I metodi di Jacobi, di Gauss-Seidel e del rilassamento. Risultati di convergenza. Criteri di arresto. Approssimazione di autovalori e autovettori: localizzazione geometrica degli autovalori. Analisi di stabilità e condizionamento. Il metodo delle potenze e delle potenze inverse. Il metodo QR. Il metodo QR per matrici in forma di Hessemberg. Riduzione di una matrice in forma di Hessemberg. Il metodo LR. Un metodo per il calcolo di autovalori di matrici simmetriche: il metodo delle successioni di Sturm.
- *Ricerca di radici di equazioni e sistemi non lineari.*
Il metodo delle iterazioni di punto fisso. Risultati di convergenza. Criteri di arresto. Metodi iterativi per sistemi non lineari.

Testi consigliati:

1. V. Comincioli. *Analisi numerica*, Springer.
2. A. Quarteroni, R. Sacco, F. Saleri. *Matematica numerica*, Springer.
3. J. Stoer. *Introduzione all'analisi numerica*, Vol. II, Zanichelli.

CALCOLO PARALLELO

Docente: docenza da assegnare

Crediti (CFU): 4

Finalità: il corso ha come obiettivi lo studio e l'analisi delle principali tecniche su cui si basa la realizzazione dei moderni sistemi di calcolo ad alte prestazioni.

Programma del corso:

- L'architettura base del processore
 - L'ISA
 - Processori multiciclo
 - L'equazione del processore
 - L'impatto della gerarchia di memoria
- Tecniche avanzate per lo sfruttamento del parallelismo a livello istruzioni (ILP)
 - Parallelismo parziale e la pipeline
 - ILP: concetti e proprietà principali
 - Le architetture superscalari
 - * Scheduling dinamico
 - * Branch prediction
 - * Issue Multipla
- Architetture a parallelismo esplicito
 - Limitazioni intrinseche dell'approccio ILP
 - Le architetture VLIW
 - Speculazione e predicazione
 - Metodi software per i processori a parallelismo esplicito
- Calcolatori paralleli
 - I paradigmi classici del calcolo parallelo
 - Tecnologie di interconnessione
 - Tecniche di programmazione
 - Architetture parallele ad uso generale e customizzate
- Un'architettura parallela per la fisica: APENext
 - Il progetto APE
 - Il calcolo su reticolo
 - Da architettura SIMD a SPMD e MIMD
 - L'architettura del processore J&T
 - L'architettura del sistema
 - Da Linux a NOSE: il sistema operativo
 - Strumenti di sviluppo

CALCOLO PROBABILITÀ E STATISTICA

Docente: G. M. Tessitore

Collaboratore didattico: F. Morandin

Crediti (CFU): 6

Programma del corso:

- Modelli probabilistici, spazi probabilizzati finiti e spazi probabilizzati numerabili, densità di probabilità. Probabilità condizionata e indipendenza. Densità binomiale e ipergeometrica. Densità di Poisson e teorema degli eventi rari.
- Lo schema di Bernoulli. Il caso finito. Il caso infinito, insufficienza degli spazi numerabili. Definizione di σ -algebra e di Probabilità σ -additive, esistenza della probabilità prodotto. Tempo di attesa del primo successo, numero di successi in n -prove, tempo di attesa dell' n -esimo successo.
- Variabili aleatorie discrete, leggi e loro densità. Densità congiunte e marginali. Indipendenza Speranza matematica. Momenti. Varianza. Disuguaglianza di Chebyshev. Covarianza e coefficiente di correlazione.
- La legge dei grandi numeri. Cenni al teorema del limite centrale.
- Catene di Markov, esempi. Classificazione degli stati, misure invarianti, teorema ergodico, problemi di assorbimento.

Testi consigliati:

1. Dispense a cura del docente.

CHIMICA

Docente: P. Cozzini

Crediti (CFU): 6

Programma del corso:

- *Introduzione.* Nomenclatura dei composti inorganici. Concetto di mole. Stechiometria chimica.
- *Struttura atomica della materia.* Orbitali atomici. Configurazione elettronica degli elementi dei blocchi s e p. Tavola periodica e proprietà periodiche.
- *Legame chimico.* Concetto di legame. Formalismo di Lewis. Formule di struttura. Legame ionico. Legame covalente. Legame metallico. Previsione della geometria molecolare. Orbitali ibridi.
- *Stati di aggregazione della materia.* Proprietà dei gas. Proprietà dei liquidi. Proprietà dei solidi.
- *Soluzioni.* Modi di esprimere le concentrazioni
- *Termochimica e termodinamica chimica.* Energetica e reazioni chimiche. Entalpia. I principi della termodinamica. Entropia. Spontaneità delle reazioni. Energia libera di Gibbs.
- *Equilibrio chimico.* Legge d'azione delle masse. Costanti d'equilibrio. Equilibri di scambio protonico. Acidi e basi. Prodotto ionico dell'acqua. pH. Idrolisi.
- *Chimica organica.* Gli idrocarburi. Gruppi funzionali. Biomolecole

Testi consigliati:

1. P. W. Atkins. *Fondamenti di Chimica*, ed. Zanichelli.

COMPLEMENTI DI ANALISI MATEMATICA

Docente: A. Lunardi

Collaboratore didattico: L. Lorenzi

Crediti (CFU): 6

Programma del corso:

- Norme e distanze. Prodotto scalare in \mathbb{R}^n e proprietà. Norma euclidea, disuguaglianza di Schwarz. Norme e spazi normati. Distanze e spazi metrici. Topologia di uno spazio metrico. Compattezza. Distanze equivalenti, norme equivalenti. Funzioni continue. Funzioni Lipschitziane. Successioni e serie negli spazi normati. Completezza, spazi di Banach. Spazi di funzioni continue, norma uniforme, convergenza uniforme di funzioni. Applicazioni lineari continue, norma degli operatori. Teorema di punto fisso delle contrazioni. Chiusi invarianti. Esempi e applicazioni scelte del teorema delle contrazioni. La serie di Von Neumann.
- Calcolo differenziale in più variabili. Differenziale di una applicazione. Somma e composizione. Varie nozioni collegate: esempi e confronti. Differenziale della applicazione inversa. Curve. Teorema del valor medio generalizzato. Lipschitzianità locale. Teorema del differenziale totale. Teorema di simmetria del differenziale secondo. Perturbazioni Lipschitziane dell'identità e omeomorfismi bilipschitziani fra aperti. Teorema di inversione locale. Teorema della funzione implicita (di Dini). Forme differenziali e campi vettoriali. Integrale di linea lungo un cammino. Forme esatte, condizioni necessarie e sufficienti per l'esattezza. Forme differenziali di classe C^2 ; forme chiuse. Domini stellati, esattezza delle forme differenziali chiuse su domini stellati.
- Massimi e minimi in più variabili. Estremi liberi: condizioni necessarie e sufficienti. Vincoli regolari. Ricerca di massimi e minimi vincolati; metodo dei moltiplicatori di Lagrange.

Testi consigliati:

1. Fleming. *Functions of several variables*, Springer.
2. Pagani–Salsa. *Analisi matematica*, Masson.

ELETTRONICA 1

Docente: docenza da assegnare

Crediti (CFU): 4

Finalità: l'elettronica digitale è alla base della tecnologia utilizzata per il progetto dei calcolatori e per le applicazioni che riguardano l'acquisizione, il controllo e la elaborazione dei segnali. Il corso intende fornire le conoscenze di base e gli strumenti per comprendere il funzionamento dei circuiti digitali.

Programma del corso:

- Segnali analogici e digitali
- Circuiti digitali
- Funzioni logiche e porte logiche
- Circuiti logici combinatori
- Algebra di Boole
- Codici binari
- Minimizzazione delle funzioni logiche
- Applicazioni di logica combinatoria:
 - Sommatore
 - Codificatori e decodificatori
 - Multiplexer e demultiplexer
- Circuiti logici sequenziali:
 - Flip flop
 - Contatori
 - Registri
- Memorie a semiconduttori
- Famiglie di circuiti integrati digitali
- Conversione analogico/digitale e digitale/analogica

ELETTRONICA 2

Docente: P. Podini

Crediti (CFU): 4

Finalità: il corso intende presentare l'utilizzo dei circuiti integrati, in particolare degli amplificatori operazionali, nella realizzazione di sistemi per la rivelazione di segnali ed il loro trattamento, non solo in modo teorico, ma attraverso l'ausilio di sessioni di Laboratorio. Prerequisiti del corso: conoscenza matematica dei vettori nel campo complesso, fondamenti di analisi circuitale e dei componenti elettronici discreti. Modalità dell'esame finale: l'esame finale consiste di una prova scritta, un'esperienza in Laboratorio e di un colloquio orale.

Programma del corso:

- Modello degli amplificatori operazionali e suoi limiti - Circuiti fondamentali a retroazione negativa: Amplificatore invertente - Sommatore - Integratore - Derivatore - Differenziale - Amplificatore non invertente - Generatore di corrente - Sfasatore - Duplicatore di frequenza - Convertitore corrente \rightarrow tensione.
- Circuiti fondamentali a retroazione positiva: Smitt trigger - Multivibratori astabili - Generatori di onde quadre e triangolari - Oscillatori sinusoidali.
- Cenni sull'utilizzo di componenti non lineari con gli amplificatori operazionali.
- Cenni sulle sorgenti di rumore

Testi consigliati:

1. P. H. Beards. *Elettronica Analogica e Digitale*, Jackson Editor.
2. F. R. Condor. *Rumore* (serie: Fondamenti di elettronica e telecomunicazioni, Vol. 6), Franco Muzzio & c. editore.

FISICA 1

Docente: M. Bonini

Crediti (CFU): 6

Finalità: il corso si propone di fornire una conoscenza organica delle leggi fondamentali della meccanica classica del punto materiale e dei sistemi. Tratta inoltre i fenomeni elettrici e magnetici fondamentali e si propone di fornire le conoscenze di base per ulteriori approfondimenti nei campi della teoria e delle applicazioni dell'elettromagnetismo. È presupposta la conoscenza dei fondamenti del calcolo differenziale ed integrale. Una parte del corso sarà dedicata ad esercitazioni in aula. Dopo aver sviluppato la teoria relativa, agli studenti saranno assegnati degli esercizi e quesiti in modo da chiarire ed approfondire gli argomenti di teoria svolti. Modalità d'esame: l'esame prevede una prova scritta integrata con una prova orale. Durante il corso sono previste delle prove in itinere che se positive potranno sostituire la prova scritta di esame finale.

Programma del corso:

- *Introduzione.*
Le grandezze fisiche. Sistemi di unità di misura. Scalari e vettori. Operazioni con i vettori: somma, prodotto scalare e prodotto vettoriale. Analisi dimensionale.
- *Cinematica del punto materiale.*
Legge oraria, velocità, accelerazione. Moto rettilineo uniforme. Moto uniformemente accelerato. Moto in due dimensioni e moto di un proiettile. Moto circolare uniforme. Accelerazione tangenziale e radiale.
- *Dinamica del punto materiale.*
Forze e secondo principio della dinamica. Legge di azione e reazione. Forze di attrito.
- *Applicazioni delle leggi della meccanica.*
Piano inclinato e reazioni vincolari. Moto armonico. Pendolo semplice.
- *Lavoro ed energia.*
Lavoro delle forze. Energia cinetica. Campi di forza conservativi. Energia potenziale. Conservazione dell'energia meccanica. Alcune forze conservative e l'energia potenziale ad esse associata.
- *Dinamica dei sistemi.*
Sistemi a molte particelle. Centro di massa. Quantità di moto e sua conservazione. Fenomeni d'urto. Problemi d'urto elastico ed inelastico.
- *Gravitazione universale.*
Leggi di Keplero. Legge di Newton. Potenziale gravitazionale. Velocità di fuga.

- *Teoria cinetica dei gas e termodinamica.*
Legge dei gas perfetti. Concetto di temperatura e funzione di stato. Conservazione dell'energia e primo principio della termodinamica. Entropia e il secondo principio della termodinamica.
- *Elettrostatica.*
Cariche elettriche e legge di Coulomb. Definizione del campo elettrico. La legge di Gauss per i campi elettrici. Applicazioni. Il potenziale elettrico. Energia potenziale elettrostatica.
- *Conduttori in equilibrio elettrostatico.*
Potenziale di un conduttore. Capacità elettrica. Condensatori. Condensatori in serie e in parallelo.
- *Corrente e circuiti a corrente continua.*
Corrente elettrica e densità di corrente. Resistenza elettrica e legge di Ohm. Forza elettromotrice. Legge di Joule. Resistenze in serie e in parallelo. Circuiti RC.
- *Fenomeni magnetici stazionari nel vuoto.*
Forza di Lorentz e definizione del campo magnetico. Forza esercitata da un campo magnetico su una corrente elettrica. Campo magnetico generato da correnti. Il campo magnetico di un filo rettilineo indefinito. La forza tra due correnti parallele. Assenza di cariche magnetiche isolate. Legge di Ampere e teorema della circuitazione. Il campo magnetico di un solenoide.
- *Campi elettrici e magnetici variabili nel tempo.*
Induzione magnetica. Legge di Faraday-Neuman. Alcuni esempi e applicazioni. Auto e mutua induzione. Corrente di spostamento. Equazioni di Maxwell.

Testi consigliati:

1. D. Halliday, R. Resnick, J. Walker. *Fondamenti di Fisica*, ed. Ambrosiana, Milano.
2. P. Mazoldi, M. Nigro, C. Voci. *Elementi di Fisica*, vol. 1 e 2, Ed. EdiSES, Napoli.

FISICA 2

Docente: L. Romanò

Crediti (CFU): 6

Finalità: gli obiettivi che si vogliono raggiungere in questo corso possono essere sintetizzati come segue:

1. conoscenza dei fondamenti dell'elettromagnetismo;
2. capacità di risolvere semplici problemi attinenti all'argomento;
3. conoscenza di applicazioni dell'elettromagnetismo classico.

Per questo, durante il corso, senza rinunciare ad un formalismo matematico rigoroso, verrà dato un grande spazio agli esercizi e alle applicazioni.

Programma del corso: il programma è articolato in due parti. Nella prima parte sono elencati gli argomenti che costituiscono il nucleo base di conoscenze di elettromagnetismo che lo studente deve acquisire. Nella seconda parte sono elencati argomenti che verranno svolti a seconda del tempo a disposizione e della risposta della classe.

ARGOMENTI BASE

- *Elettrostatica nel vuoto:*
carica elettrica, legge di Coulomb, campo elettrico, teorema di Gauss e 1^a equazione di Maxwell, potenziale elettrico, dipolo elettrico, conduttori, capacità elettrica, sistemi di condensatori, collegamento in serie e in parallelo, energia del campo elettrostatico.
- *Corrente elettrica stazionaria:*
resistenza elettrica e legge di Ohm, effetto Joule, forza elettromotrice e generatori elettrici, circuiti in corrente continua.
- *Magnetismo nel vuoto:*
forza di Lorentz, vettore induzione magnetica, forze magnetica su una corrente, momento magnetico della spira percorsa da corrente, relazione tra momento meccanico e momento magnetico, campi generati da correnti stazionarie, legge di Biot e Savart (campo del filo indefinito, della spira circolare e del solenoide), 2^a equazione di Maxwell, teorema di Ampère.
- *Campi magnetici variabili nel tempo:*
induzione elettromagnetica, legge di Faraday-Newmann, 3^a e 4^a equazione di Maxwell, autoinduzione, circuito RL, energia magnetica.
- *Onde:*
equazione d'onda, tipi di onde, velocità di fase, equazioni delle onde elettromagnetiche e loro proprietà, onda piana e onde sferiche, energia di un'onda elettromagnetica e vettore di Poynting, spettro della radiazione elettromagnetica.

ARGOMENTI AGGIUNTIVI

- *Elettrostatica nella materia:*
la costante dielettrica, interpretazione microscopica, suscettibilità elettrica.
- *Magnetismo nella materia:*
vettori B, H e M, materiali paramagnetici, ferromagnetici, diamagnetici, legge di Curie, ciclo di isteresi.
- *Polarizzazione della luce:*
birifrangenza e lamine polarizzanti.
- *Ottica:*
principio di Fermat, riflessione e rifrazione (specchi e lenti), interferenza (esperimento di Young); diffrazione e potere risolutivo.

Testi consigliati:

1. C. Mencuccini, V. Silvestrini. *Fisica II: Elettromagnetismo Ottica*, Liguori editore.

FONDAMENTI DELL'INFORMATICA

Docente: R. Bagnara

Collaboratore didattico: E. Zaffanella

Crediti (CFU): 6

Programma del corso:

- Cenni introduttivi sul concetto di algoritmo, sulla rappresentazione dell'informazione, e sull'architettura del calcolatore.
- Linguaggi formali.
- Espressioni regolari.
- Automi a stati finiti.
- Grammatiche generative.
- Linguaggi liberi dal contesto.
- Macchine di Turing.
- Funzioni calcolabili e non.

Testi consigliati:

1. A. Dovier, R. Giacobazzi. *Fondamenti dell'Informatica: Linguaggi Formali e Calcolabilità*.
2. A. M. Pitts. *Regular Languages and Finite Automata*.
3. I. Mastroeni. *Eserciziario per il corso "Fondamenti dell'Informatica: Linguaggi Formali e Calcolabilità"*.

FONDAMENTI DI PROGRAMMAZIONE (+ LABORATORIO)

Docente: G. Rossi

Docente per il laboratorio: G. Rossi

Crediti (CFU): 6 per la parte di Fondamenti, 4 per il Laboratorio

Programma del corso:

PARTE I - Programmazione imperativa con il C++

- *Concetti e costrutti di base.*

Il concetto di algoritmo. Rappresentazione di algoritmi: i diagrammi di flusso.
I linguaggi di programmazione. Esecuzione tramite compilazione ed interpretazione
Struttura di un programma. Un esempio di programma C++.
Variabili e costanti. Dichiarazione di variabile.
Tipi di dato. I tipi elementari del C++.
Assegnamento e statement composto.
Input/output di base: formattato, a caratteri. Gli stream del C++.
Espressioni e operatori.
Le strutture di controllo: selezione, cicli, salto. Gli statement del C++.
Strutture dati: definizione, costruttori di tipo, strutture dati astratte e concrete.
Vettori e matrici. Realizzazioni in C++ tramite array. Array bidimensionali.
Stringhe. Stringhe come array di caratteri in C++. Operazioni su stringhe.
Funzioni di libreria.
Il costruttore `struct` in C++: dichiarazione, selezione elementi, operazioni. Record e tabelle (cenni).
La dichiarazione `typedef`.
- *Funzioni ed astrazione procedurale.*

Sottoprogrammi: significato, motivazioni. Dichiarazione e chiamata di funzione in C++.
Modalità di passaggio parametri: per valore, per riferimento.
Passaggio parametri di tipo array in C++.
Funzioni `void`.
Funzioni ricorsive.
Visibilità identificatori: regole di “scope” in C++. Dichiarazioni globali e locali.
- *Sviluppo programmi.*

Ambiente di sviluppo: editor, compilatore, linker (cenni).
Sviluppo di semplici programmi nel frammento imperativo del C++.
Input/output su file. Creazione di uno stream. Lettura/scrittura formattata e a caratteri.

Utilizzo di funzioni di libreria.

PARTE II - Programmazione orientata agli oggetti con il C++

- *Concetti e costrutti di base.*
Dichiarazione di classe. Creazione di oggetti. Accesso ai campi di un oggetto.
Funzioni proprie ed invocazione di funzioni.
Inizializzazione di oggetti: costruttori.
Esempio: classe `Razionale`.
Overloading di funzioni ed operatori. Ridefinizione operatori `>>` e `<<`.
La classe `string`.
Puntatori: dichiarazione, accesso, operazioni.
Allocazione dinamica della memoria. Puntatori ed array.
Esempio: classe `int_vett`. Ridefinizione `operator []`.
Ridefinizione assegnamento e costruttore di copia. Distruttori.
Tempo di vita di un oggetto (cenni).
Tipi di dato astratti (ADT). “Incapsulamento” e “information hiding”.
- *Concetti e costrutti avanzati.*
Gestione delle eccezioni. Motivazioni. Costrutti e meccanismi del C++.
Classi parametriche: dichiarazione, istanziamento. Esempio: classe `vett<t>`.
Funzioni parametriche (cenni).
Ereditarietà: dichiarazione, ridefinizione funzioni e dati. Campi `protected`.
Costruttori e distruttori in presenza di ereditarietà. Modalità di derivazione.
Ereditarietà multipla (cenni).
- *Sviluppo programmi.*
Librerie standard del C++ (cenni).
Sviluppo di semplici programmi in C++.
Progettazione ed implementazione di un’applicazione in C++.

Testi consigliati:

1. Per la Parte I:
 - R. Miller, D. Clark, B. White, E. W. Knottenbel. *An Introduction to the Imperative Part of C++*, 1999,
2. Per l’intero corso:
 - H. M. Deitel, P. J. Deitel. *C++ Fondamenti di programmazione*, Apogeo, 2003, 696 pp. (il più “didattico”)
 - S. Lippman, J. Lajoie. *C++: Corso di programmazione*, 3a Ed., Addison-Wesley, 2000. (completo, ma un po’ più “ostico” del precedente)
 - B. Stroustrup. *Il linguaggio C++*, Addison-Wesley. (dal “padre” del linguaggio: completo e preciso, ma piuttosto “ostico”)

GEOMETRIA

Docente: docenza da assegnare

Crediti (CFU): 6

Programma del corso:

- Introduzione e studio degli spazi affini ed affini euclidei.
- Geometria analitica in tali spazi.
- Elementi impropri ed ampliamento proiettivo degli spazi affini e affini euclidei.
- Complessificazione degli spazi affini e affini euclidei reali.
- Generalità sulle curve.
- Coniche negli spazi affini e affini euclidei.

Testi consigliati:

1. V. Mangione. *Spazi Curve e Superficie*, Azzali, 1998.
2. M. Stoka, *Esercizi di Geometria*, CEDAM, 1995.

GRAFICA COMPUTAZIONALE TECNICA

Docente: M. Bordegoni

Crediti (CFU): 5

Finalità: il corso si propone di fornire all'allievo conoscenze su metodi e tecniche per la generazione, modifica e gestione di modelli geometrici da utilizzarsi come supporto per la creazione di scene 3D e/o 4D in applicazioni di grafica statiche o dinamiche. Il corso prevede l'utilizzo di un modellatore geometrico e di una libreria grafica e di animazione per la creazione di scene 3D e 4D. L'esame prevede la realizzazione di una applicazione di grafica 3D o 4D assegnata, e di una prova orale relativa al programma svolto nel corso delle lezioni.

Programma del corso:

- Introduzione alla Modellazione Geometrica e alla Computer Graphics.
- Metodi e tecniche di Modellazione Geometrica.
- Modellazione di solidi e di superfici.
- Formati di dati.
- Tecniche di manipolazione di modelli geometrici.
- Gestione del viewing.
- Tecniche di rendering e texturing.
- Tecniche di animazione e di interazione.
- Librerie di modellazione geometrica e librerie grafiche.

Testi consigliati:

1. J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes. *Computer Graphics: principle and practice in C*, Addison-Wesley, 1997.
2. M. E. Mortenson. *Geometric Modeling*, Ed. John Wiley & Sons, 1997.
3. M. O'Rourke. *Principles of three-dimensional computer animation*, Norton & Co., 1998.

INFORMATICA IN AZIENDA

Docente: G. Destri

Crediti (CFU): 3

Finalità: acquisire le conoscenze relative alle tecnologie in uso nelle aziende e alla organizzazione dell'ICT entro esse, anche relativamente alla gestione della sicurezza.

Programma del corso:

- *L'informatica "come servizio".*
Il concetto di sistema informativo ed il ruolo dell'ICT in esso.
L'organizzazione dell'informatica aziendale, con definizione di professionalità e ruoli.
I concetti base di rete; le reti in azienda.
- *La strutturazione a livelli logici per le applicazioni.* La stratificazione dell'ICT aziendale: logico-funzionale; spaziale; storica. Il problema della retro-compatibilità.
Gli applicativi utente; gli applicativi gestionali; gli applicativi enterprise: client-server e ERP; i database server; groupware server.
L'infrastruttura: domain management; i sistemi operativi in uso nelle aziende; le reti aziendali.
- *Come gestire il tutto.*
Sistemisti ed utenti.
Le tecnologie: linguaggi, ambienti, sistemi operativi.
- *I dati: il "tesoro" di ogni azienda.*
Introduzione ai concetti di sicurezza: Safety e Security; soluzioni tecniche per la Safety; i pericoli per i sistemi.
Classificazione dei tipi di attacco: intrusione, intercettazione, impersonificazione, denial of service; il ruolo dei Virus.
- *La protezione dei dati.*
L'identità elettronica e il controllo degli accessi.
La gestione globale dei sistemi: le politiche di gestione; sistemi ed utenti.
Il rapporto costi-benefici.
"Quis custodiet custodem?" .

Testi consigliati:

1. Dispense a cura del docente.

INFORMATICA TEORICA

Docente: G. Ricci

Crediti (CFU): 5

Programma del corso:

- Complementi calcolabilità Turing: enumerazione e forma normale di Kleene.
- Sistemi termini combinatorii: sistema S-K-I e sistemi alternativi.
- Rappresentabilità combinatoria: caso funzioni primitive ricorsive e cenno caso funzioni parziali ricorsive.
- Inseparabilità ricorsiva: teorema Curry e 3 corollari.
- Estensionalità: formulazioni alternative.
- Tipi: caso termini con tipo, caso assegnazione, stratificabilità, riduzione soggetto e cenno caso generale.
- Applicazioni matematiche ed informatiche: rappresentazione algebre basate, generazione sistemi Menger e monoidi analitici.

Testi consigliati:

1. J. R. Hindley, B. Lercher, J. P. Seldin., *Introduzione alla Logica Combinatoria*, Boringhieri, 1974.
2. H. Jr. Rogers. *Theory of Recursive Functions and Effective Computability*, McGraw-Hill, 1967.

INGEGNERIA DEL SOFTWARE

Docente: docenza da assegnare

Crediti (CFU): 6

Programma del corso:

- *Il processo di sviluppo del software:* aspetti economici, organizzativi e metodologici; prodotto software e processo; il ciclo di vita dei sistemi software; modello tradizionale a cascata; modello evolutivo e a fontana.
- *Analisi e specifica dei requisiti:* modelli e linguaggi di specifica; UML ed il suo uso.
- *Progettazione dei sistemi software:* principi e metodi di progettazione; principi di modularità ed incapsulamento; la progettazione orientata agli oggetti; i “design patterns” ed il loro uso; architetture multi-tier e Web.
- *Realizzazione dei sistemi software:* linguaggi di programmazione ad oggetti; il linguaggio Java; l’architettura di Java2; Java e le proprietà di oggetti; sistemi multi-tier in Java; sistemi Web in Java; cenni al linguaggio C# e all’architettura .NET.
- *La conduzione operativa di un progetto:* impostazione e definizione di obiettivi; analisi dei vincoli; scelta di strumenti e architetture; il lavoro in team; test di regressione; sviluppi parziali e test parziali; test di aggregazione; test su dati reali; collaudo; entrata in produzione e manutenzione ordinaria; manutenzione evolutiva; case study.
- *Metodologie di gestione dei progetti software:* il project management; metriche e diagrammi utili (Gantt, Perth, . . .); strumenti CASE; il problema della documentazione; comunicazione entro e fuori un team; evoluzione e manutenibilità dei sistemi.

INGLESE

Crediti (CFU): 3

Lezioni ed esame di idoneità: le lezioni e gli esami per il corso di Lingua Inglese sono tenuti presso il Campus dal personale del *Centro Linguistico di Ateneo*. La partecipazione all'esame è permessa in tutte le sessioni di esame (previa iscrizione all'appello da effettuarsi utilizzando la procedura telematica), indipendentemente dal fatto che il corso sia indicato come insegnamento del secondo semestre.

Riconoscimento titoli in possesso dello studente: secondo il protocollo d'intesa firmato dal MIUR (Ministero dell'Istruzione, dell'Università e della Ricerca) e dalla CRUI (Conferenza dei Rettori delle Università Italiane), e sulla base delle indicazioni del Concilio d'Europa, il titolo di *Preliminar English Test* (PET) è riconosciuto come attestato di "idoneità" per gli esami di primo livello. Sono altresì riconosciuti, essendo titoli di livello superiore al suddetto, i seguenti: *First Certificate in English* (FCE), *Certificate of Advanced English* (CAE), *Certificate of Proficiency in English* (CPE) e *Test of English as a Foreign Language* (TOEFL). Gli studenti in possesso di uno dei titoli suddetti possono ottenere l'idoneità presentandosi al Centro Linguistico con il certificato originale e consegnando una fotocopia dello stesso ed una fotocopia del frontespizio del libretto universitario: in tal modo i loro nominativi verranno automaticamente inseriti nell'elenco degli studenti idonei alla prima data di esame successiva alla consegna della documentazione.

Per ulteriori informazioni:

Centro Linguistico di Ateneo
Parco Area delle Scienze 45/a
43100 – Parma
Tel.: +39 0521 905578 – Fax: +39 0521 905580
E-mail: c.ling@ipruniv.cce.unipr.it
Web: <http://www.unipr.it/arpa/cla/>

Telelingua: Il Corso di Laurea in Informatica aderisce a *Telelingua*, un'iniziativa proposta nell'ambito del *Progetto Campus One – Abilità Linguistiche* che si ripropone di sperimentare nuove metodologie di insegnamento linguistico e di introdurre sistemi di certificazione linguistica riconosciuti a livello internazionale. Per maggiori informazioni: <http://stream.cedi.unipr.it/main/index.php>

INTELLIGENZA ARTIFICIALE

Docente: D. Bianchi

Crediti (CFU): 6

Programma del corso:

- *Soluzione di problemi.*
Soluzione di problemi basata sulla ricerca nello spazio degli stati. Ricerca in ampiezza e ricerca in profondità. Metodi di ricerca informata: algoritmo A^* . Risoluzione di giochi basata sulla ricerca: algoritmo minimax e potatura alfa-beta.
- *Rappresentazione della conoscenza.*
Logica dei predicati. Clausole. Risoluzione. Programmazione logica e Prolog.
- *Conoscenze strutturate.*
Reti semantiche e frame. Reti ad ereditarietà strutturata. Ontologie e applicazioni al Web semantico.
- *Rappresentazione di conoscenza incerta.*
Probabilità a priori e condizionata, regola di Bayes, reti bayesiane, logica fuzzy.
- *La pianificazione.*
Caratteristiche generali di un sistema di pianificazione. Il mondo dei blocchi. STRIPS. Pianificazione gerarchica. Pianificazione nel mondo reale: pianificazione condizionale e controllo dell'esecuzione.
- *L'apprendimento.*
Apprendimento induttivo: alberi di decisione. Apprendimento per rinforzo. Apprendimento e conoscenza. Programmazione logica induttiva. Apprendimento mediante esplorazione. Reti neurali. Algoritmi genetici ed evolutivi.
- *Elaborazione del linguaggio naturale.*
Linguaggi naturali e formali. Grammatiche. Analisi sintattica. Ambiguità. Rappresentazione semantica. Comprensione del discorso. Applicazioni. Riconoscimento del parlato. Modelli Markov. Generazione del parlato.
- *Agenti.*
Agenti e i sistemi multi-agente cooperanti e concorrenti. Comunicazione fra agenti e atti linguistici. Modelli e architetture di agente.

Testi consigliati:

1. Stuart Russell e Peter Norvig. *Artificial Intelligence: A Modern Approach* (seconda edizione, in inglese). Prentice Hall, 2003.

2. Stuart Russell e Peter Norvig. *Intelligenza artificiale: un approccio moderno* (traduzione italiana della prima edizione). UTET Libreria, 1998. A cura di Luigia Carlucci Aiello.
3. N. J. Nilsson. *Intelligenza Artificiale*, Apogeo 2002.

LABORATORIO DI GEOINFORMATICA

Docente: R. Valloni

Crediti (CFU): 5

Programma del corso:

- Caratteristiche ed estensioni di ArcView
- Georeferenziazione di dati territoriali e cartografie raster
- Acquisizione di informazione da fonti diverse e integrazione in un SIT
- Progettazione e realizzazione di basi numeriche
- Misure e interpretazioni su ortofotografie
- Editing di temi territoriali a scala nazionale
- Rappresentazione del sottosuolo con isolinee
- Rappresentazione di superfici geologiche del sottosuolo
- Modellazione fisica di corpi geologici del sottosuolo padano
- Trattamento e analisi d'immagini: misure direzionali e misure areali a varia scala su corpi sedimentari.

LAMBDA CALCOLO E COMBINATORI

Docente: G. Ricci

Crediti (CFU): 4

Programma del corso:

- Lambda termini: riduzioni, teoremi Church-Rosser e corollari.
- Sistemi equivalenti: alberi DeBruijn, sistema Combinatori e cenni LISP.
- Rappresentazione ricorsività: iteratori, diadi, ricorsore primitivo e cenni teoremi rappresentazione.
- Indecidibilità: inseparabilità ricorsiva, corollari e cenni decidibilità sistema con tipi.

Testi consigliati:

1. J. R. Hindley, J. P. Seldin. *Introduction to Combinators and λ -Calculus*, Cambridge University Press, 1986.
2. J. R. Hindley, B. Lercher, J. P. Seldin. *Introduzione alla Logica Combinatoria*, Boringhieri, 1974.

LINGUAGGI DICHIARATIVI

Docente: G. Rossi

Crediti (CFU): 4

Programma del corso:

- Programmazione dichiarativa: caratteristiche, vantaggi, linguaggi di programmazione.
- Il linguaggio Prolog.
Sintassi: termini, clausole, programma. Semantica dichiarativa informale. Interpretazione procedurale. Unificazione e sostituzioni
Semantica operativa. Derivazione. Albero di derivazione SLD. Insieme di successo.
Non-determinismo e strategie di ricerca. Backtracking. Incompletezza.
Rappresentazione e manipolazione liste – Definizioni ricorsive.
Rappresentazione e manipolazione numeri. Soluzione “logica” ed “extra-logica”. Il predicato `is` – Funzioni vs. relazioni.
Predicati built-in. Dalla Programmazione Logica al Prolog.
Controllo backtracking (“cut”) – Negazione per fallimento (cenni).
Input-output di base e da file.
Stringhe e atomi. Input-output di stringhe.
Modifica dinamica del programma.
Raccolta dei risultati (`setof`).
- Linguaggi di programmazione logica a vincoli. Motivazioni. Nozioni di base. Casi di studio: vincoli su “domini finiti” e su insiemi.
- Laboratorio.
L’ambiente di programmazione SWI-Prolog.
Sviluppo ed esecuzione semplici programmi Prolog.

Testi consigliati:

- L. Console, E. Lamma, P. Mello: *Programmazione logica e Prolog (II edizione)*, UTET Libreria, 1997.

LOGICA MATEMATICA

Docente: M. Servi

Collaboratore didattico: C. Reggiani

Crediti (CFU): 6

Programma del corso:

- Un calcolo dei sequent per la logica proposizionale classica. Linguaggi del primo ordine e deduzione naturale predicativa.
- Semantica. Insiemi non contraddittori e insiemi soddisfacibili di formule. Varie formulazioni equivalenti del teorema di completezza e i vari metateoremi classici come suoi corollari.

Testi consigliati:

1. H. Enderton. *A Mathematical Introduction to Logic*, Academic Press, 1972.
2. W. S. Hatcher. *Fondamenti della matematica*, Boringhieri, 1973.
3. E. Mendelson. *Introduzione alla logica matematica*, Boringhieri 1972.
4. C. Reggiani, M. Servi. *Lezioni di Logica Matematica*, 1, Libreria S. Croce, 2002.

METODOLOGIE DI PROGRAMMAZIONE

Docente: E. Zaffanella

Collaboratore didattico: R. Bagnara

Crediti (CFU): 6

Finalità: la programmazione orientata agli oggetti si fonda su alcuni principi (incapsulamento, ereditarietà, polimorfismo, ecc.) che l'esperienza ha mostrato essere fondamentali per lo sviluppo di software chiaro, conciso, riutilizzabile e di facile manutenzione. I linguaggi di programmazione più diffusi rendono disponibili alcuni strumenti e tecniche che portano allo sviluppo di codice aderente ai principi suddetti. Il corso si propone di presentare le caratteristiche avanzate del linguaggio di programmazione C++, mostrando come un loro utilizzo corretto e consapevole porti al raggiungimento degli obiettivi preposti.

Programma del corso:

- Richiami sul linguaggio C++: tipi di dato; espressioni; istruzioni; funzioni; campo d'azione e ciclo di vita; funzioni sovraccaricate; classi.
- Cenni sulla programmazione per contratto: pre-condizioni, post-condizioni ed invarianti di classe.
- Inizializzazione, assegnamento e distruzione.
- Eccezioni e gestione delle risorse; tecniche per la gestione dinamica della memoria.
- Ereditarietà semplice: differenze tra contenimento, ereditarietà privata ed ereditarietà pubblica.
- Polimorfismo dinamico: funzioni virtuali; il principio di sostituzione di Liskov.
- Progetto di interfacce software: tipi concreti, tipi astratti, classi di interfaccia e classi implementative; ereditarietà multipla e virtuale.
- Polimorfismo statico: template di funzione e template di classe.
- La libreria standard STL: contenitori, iteratori ed algoritmi generici; oggetti funzione.
- Ambiente di sviluppo: il compilatore g++; il debugger gdb; automazione del processo di compilazione: make; controllo delle versioni: cvs.
- Documentazione di interfacce software: doxygen.

Testi consigliati:

1. B. Stroustrup. C++: *Linguaggio, libreria standard, principi di programmazione*, terza edizione, Addison-Wesley, 2000.
2. S. Lippman, J. Lajoie. C++: *Corso di programmazione*, terza edizione, Addison-Wesley, 2000.

MODELLAZIONE E SIMULAZIONI NUMERICHE

Docente: F. Di Renzo

Crediti (CFU): 4

Finalità: il corso si propone di fornire una introduzione elementare a tecniche di modellizzazione e simulazione numerica di utilizzo corrente in Fisica Computazionale. Queste tecniche, per quanto motivate e sviluppate nell'alveo di problemi di Fisica, forniscono in realtà un linguaggio generale, che non a caso ha trovato (anche in anni recenti) applicazioni a campi assai svariati (solo per citarne alcuni: economia ed analisi di mercati finanziari, reti di calcolatori, biofisica computazionale). Proprio per questo, il corso si propone di avere un carattere in larga parte seminariale: dopo una agile parte introduttiva, si svilupperà su un progetto da concordare fra docente e studenti. La prova finale consisterà appunto nella messa a punto di tale progetto.

Programma del corso:

- Richiami di probabilità e statistica. Variabili aleatorie con distribuzione assegnata. Il caso della distribuzione piatta e la generazione di successioni di numeri pseudocasuali. La distribuzione gaussiana. Tecniche generali per la generazione di successioni a fissata distribuzione di probabilità. Il metodo Montecarlo statico come tecnica di integrazione su spazi a dimensioni elevate.
- Il linguaggio della analisi degli errori. Analisi di campioni sperimentali. Il metodo di bootstrap. Cenni al problema del cosiddetto data mining.
- Introduzione alle equazioni differenziali stocastiche. Il caso del moto browniano libero e sottoposto ad una forza esterna: breve storia della equazione di Langevin. Cenni ad applicazioni dell'equazione di Langevin a contesti diversi.
- Catene di Markov e metodo Montecarlo dinamico. Simulazioni di meccanica statistica. Dinamica molecolare e sue applicazioni.
- Scelta di un progetto di simulazione (da concordare fra docente e studenti). Possibili campi di applicazione potrebbero essere:
 - la cosiddetta econofisica (applicazione di metodi della fisica statistica alla analisi finanziaria);
 - meccanica statistica e teoria dei grafi applicate a reti di calcolatori (internet) oppure a reti cosiddette sociali (rientrano in questo caso ad esempio i modelli epidemiologici);
 - meccanica statistica e genetica (elementare introduzione all'utilizzo di metodi statistici al problema della codifica di informazione genetica).

Il corso prevede di essere in larga parte condotto in laboratorio. L'ambiente privilegiato per la trattazione numerica dei problemi sarà Matlab.

Testi consigliati:

1. Appunti a cura del docente.

MODELLI E SISTEMI DINAMICI

Docente: M. Groppi

Crediti (CFU): 6

Finalità: scopo del corso è lo studio di alcuni modelli matematici di evoluzione provenienti dalla Meccanica e più in generale dalle Scienze Applicate.

Programma del corso:

- Sistemi dinamici: definizioni e proprietà elementari. Il concetto di stabilità. Metodi di Liapunov per lo studio della stabilità di soluzioni stazionarie.
- Modelli lineari: dall'oscillatore armonico ai problemi di risonanza.
- Modelli in dinamica delle popolazioni: il modello preda-predatore, il modello epidemiologico.
- Oscillatori non lineari: l'equazione di Van der Pol, l'equazione di Duffing; biforcazioni, cicli limite.
- Sistemi dinamici discreti: mappa di Feigenbaum; biforcazioni di periodo doppio.

Testi consigliati:

1. G. L. Caraffini, M. Iori, G. Spiga. *Proprietà elementari dei sistemi dinamici, Appunti per il corso di Meccanica Razionale*, Università degli Studi di Parma, a.a 1998-99.
2. G. Borgioli. *Modelli Matematici di evoluzione ed equazioni differenziali*, Quaderni di Matematica per le Scienze Applicate/2, CELID, Torino, 1996.
3. R. Riganti. *Biforcazioni e Caos nei modelli matematici delle Scienze applicate*, Levrotto & Bella Torino, 2000.
4. M. W. Hirsch, S. Smale. *Differential Equations, Dynamical Systems and Linear Algebra*, Academic Press, New York, 1974.
5. J. D. Murray. *Mathematical Biology*, Springer-Verlag, New York, 1989.
6. J. Guckenheimer, P. Holmes. *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*, Springer-Verlag, New York, 1983.

MODELLISTICA MOLECOLARE

Docente: P. Cozzini

Crediti (CFU): 4

Programma del corso:

- Banche dati per i chimici (CSD, PDB, ICSD).
- La chimica e i modelli.
- I metodi per il Drug Design.
- Metodi di Meccanica Molecolare:
 - minimizzazioni;
 - il problema dei force fields;
 - analisi conformazionale;
 - dinamica;
 - docking (manuale e automatico, funzioni di scoring);
 - calcolo di proprietà geometriche;
 - simulazione di proprietà chimico fisiche.
- Metodi Quantomeccanici:
 - metodi semiempirici;
 - metodi ab initio;
 - metodi DFT.
- Modelli nei tre stati
- L'uso di tecniche sperimentali (XRD, Polveri, NMR, IR) e modelli molecolari in feed back.
- L'implementazione dei metodi in prodotti commerciali e in software per la ricerca scientifica.

Il modulo prevede circa 30 ore di lezione frontale e le rimanenti di lezione frontale in laboratorio per la preparazione del lavoro autonomo da svolgere con la supervisione di qualcuno.

Esercitazioni:

- Molecular building “de novo” e da banche dati strutturali.
- Il problema dell'energia: minimizzazione di piccole molecole organiche.
- Analisi conformazionale di piccole molecole organiche, organometalli e piccoli peptidi.
- Calcolo di proprietà molecolari.

- Interazioni host guest: docking manuale e guidato, il problema delle funzioni di scoring in diverse classi di molecole.
- Le interfacce grafiche.

RETI DI CALCOLATORI

(+ LABORATORIO DI PROGRAMMAZIONE DI RETE)

Docente: D. Bianchi

Docente per il laboratorio: R. Alfieri

Collaboratore didattico: R. Covati

Crediti (CFU): 6 per la parte teorica, 4 per il laboratorio

Finalità: introduzione alle principali problematiche e ai principi di progettazione delle reti di calcolatori, dei protocolli e dei servizi. L'esame (integrato con quello di Laboratorio di programmazione di rete) consisterà in un accertamento delle conoscenze teoriche e nello sviluppo di un progetto.

Programma del corso:

- *Introduzione alle reti di calcolatori.*
Topologie e tipi di reti: MAN, LAN, WAN. Protocolli e servizi. Prestazioni. Modello ISO/OSI. Protocolli TCP/IP. Internetworking.
- *Livello fisico e livello data-link.*
Mezzi trasmissivi. Controllo dell'errore. Gestione del flusso. Protocolli a finestra scorrevole.
- *Livello rete e livello MAC.*
Algoritmi di routing statici e dinamici. Controllo della congestione e del flusso. Protocollo IP. Reti LAN. Ethernet, token ring. Reti ATM.
- *Livello trasporto.*
Protocolli, buffering, controllo del flusso e congestione. Multiplexing. Protocolli TCP e UDP. Esempi di applicazione.
- *Livello di applicazione.*
Protocolli HTTP, FTP, formato MIME e protocolli di accesso alla posta.
- *Sicurezza nelle reti.*
Crittografia a chiave simmetrica e a chiave pubblica. Autenticazione. Firma digitale.

Programma per la parte di Laboratorio:

- Paradigmi per la programmazione di rete.
- I socket: socket Tcp, socket Udp, RPC, programmazione in C e Java.
- La sicurezza delle reti: tipi di attacchi e contromisure, firewall, principi di crittografia applicata, autenticazione, i protocolli SSL e TLS.
- Programmazione distribuita: i Web Services, le specifiche WSDL ed il protocollo SOAP, grid computing.

Testi consigliati:

1. L. P. Peterson, B. S. Davie. *Reti di calcolatori*, Apogeo, 2004.
2. J. F. Kurose, K. W. Ross. *Internet e reti di calcolatori*, McGraw Hill, 2003.
3. A. S. Tanenbaum. *Reti di computer*, Utet Libreria, 1997.

RETI LOGICHE

Docente: S. Caselli

Crediti (CFU): 5

Finalità: fornire le basi culturali per l'analisi e la progettazione di sistemi digitali sincroni. Il corso presenterà sia le metodologie tradizionali, sia tecniche di ottimizzazione di tipo algoritmico/euristico, tipicamente presenti nei flussi di progettazione legati agli ambienti CAD di Design Automation per uso industriale. Attività d'esercitazione: le esercitazioni hanno lo scopo di familiarizzare lo studente con le problematiche di analisi delle specifiche di progetto e nel contempo acquisire padronanza delle tecniche di analisi e sintesi di circuiti combinatori e sequenziali. Modalità d'esame: due prove intermedie (modo suggerito) oppure una prova scritta complessiva. Gli scritti contengono sia esercizi sia domande di teoria. La prova orale è facoltativa e riservata solo a chi ha superato sufficientemente lo scritto.

Programma del corso:

- *Introduzione ai sistemi digitali.* Evoluzione delle tecnologie elettroniche, finalità e limiti dei sistemi digitali sincroni
- *Reti combinatorie.*
 1. Richiami: Espressioni canoniche e generali; Analisi e sintesi di funzioni completamente specificate mediante mappe di Karnaugh.
 2. Estensione delle tecniche di analisi e sintesi per reti combinatorie a due livelli: Funzioni incompletamente specificate. Reti a più uscite (metodo degli implicanti/implicati primi multipli). Analisi e sintesi di reti a NAND e a NOR.
 3. Strumenti CAD per la sintesi di reti combinatorie: Algoritmo di Quine-McCluskey. Espresso. Simulazione logica.
 4. Reti a più livelli e progettazione con moduli integrati: Fattorizzazione e scomposizione di espressioni. Progettazione mediante composizione di moduli combinatori MSI e LSI.
 5. Logiche programmabili (PLA, PAL).
 6. Circuiti combinatori dedicati: Circuiti aritmetici (sommatore, comparatore, ALU). Convertitori di codice. Circuiti per controllo di parità e codice di Hamming. Circuiti a EXOR.
 7. Fenomeni transitori nelle reti combinatorie: alee statiche e dinamiche.
- *Reti sequenziali sincrone.*
 1. Modelli di Mealy e di Moore. Analisi di circuiti logici elementari con ritardi e retroazione. Funzionamento in modo fondamentale.
 2. Reti per la memorizzazione dello stato: Latch SR e D; Flip-Flop D, JK e T. Problemi di temporizzazione.

3. Automi a stati finiti: Strumenti di definizione dell'automa (diagramma degli stati, tabelle di flusso e linguaggi di descrizione). Minimizzazione degli stati.
 4. Procedimenti di analisi e di sintesi delle reti sequenziali sincrone: Codifica degli stati. Marcatura dello stato con diversi tipi di elementi di ritardo.
 5. Comandi di Preset e Clear nei Flip-Flop sincroni. Reti con ingressi asincroni o impulsivi.
 6. Progettazione di contatori binari, contatori Johnson, registri paralleli e seriali.
 7. Logiche programmabili sequenziali (FPGA).
- *Analisi e sintesi di sistemi complessi.*
 1. Progettazione di reti sequenziali con moduli integrati (registri, contatori, registri a scorrimento).
 2. Suddivisione tra parte di controllo e datapath.
 3. Cenni al pipelining.
 4. Cenni ai formalismi per la descrizione dell'hardware.

Testi consigliati:

1. Dispense e copie delle trasparenze presentate a lezione (reperibili sul sito web), oltre ad uno tra i libri sotto indicati.
2. M. M. Mano, C. R. Kime. *Reti Logiche*, Addison-Wesley/Pearson Education Italia, 2002.
3. M. M. Mano. *Digital Design*, 3/e, Prentice Hall, 2002.

RICERCA OPERATIVA

Docente: docenza da assegnare

Crediti (CFU): 6

Programma del corso:

- *Introduzione alla Programmazione Matematica.*
- *Problemi di Programmazione Lineare:* modelli costruiti a partire da problemi reali; risultati teorici (in particolare il teorema fondamentale della PL); il metodo del simplesso con i suoi passi principali; interpretazione geometrica e algebrica del metodo del simplesso; teoria della dualità con i teoremi fondamentali che legano le risoluzioni dei due problemi primale e duale; il metodo del simplesso duale; analisi di sensitività (analisi di quanto le soluzioni finali siano sensibili a variazioni dei dati dei problemi).
- *Programmazione lineare intera:* aspetti teorici ed in particolare i legami tra un problema di PLI ed il suo rilassamento lineare; brevissimi cenni di complessità; metodi di risoluzione; algoritmi di taglio ed in particolare tagli di Gomory; algoritmi di tipo branch-and-bound.
- *Graft:* definizioni di base ed algoritmi per risolvere semplici problemi su di essi come l'individuazione di un albero di supporto a costo minimo.
- (eventuale) *Il problema di flusso massimo su reti:* modello matematico e algoritmo di Ford-Fulkerson

Testi consigliati:

1. Dispense a cura del docente.

SCRITTURA MATEMATICA E INFORMATICA

Docente: R. Bagnara

Collaboratore didattico: A. Zaccagnini

Crediti (CFU): 3

Finalità: il corso si propone di introdurre le tecniche della comunicazione scritta in ambito matematico e informatico. L'obiettivo è quello di mettere gli studenti in grado di presentare matematica e informatica in maniera chiara, convincente, gradevole: in una parola, efficace. Il corso affronta questo problema da vari punti di vista: l'uso di tecniche espressive differenti a seconda del tipo di scritto e dei destinatari del medesimo, gli elementi dello stile, la tipografia matematica, l'uso dei linguaggi di composizione TeX e LaTeX, e del linguaggio grafico MetaPost.

Nelle esercitazioni e per sostenere l'esame finale gli studenti lavoreranno su esempi concreti, scrivendo nuovi testi e leggendone criticamente (eventualmente correggendoli) altri.

Programma del corso:

- L'argomentazione: in generale e in matematica e informatica.
- Scrivere cosa e per chi:
 - Il testo didattico.
 - Il pezzo divulgativo.
 - La dissertazione di laurea.
 - La documentazione di software per i programmatori.
 - La documentazione di software per gli utenti.
 - L'articolo scientifico.
- Gli elementi dello stile:
 - Grammatica e composizione.
 - La punteggiatura.
 - Riferimenti bibliografici.
 - Citazioni.
- Tipografia matematica:
 - La composizione delle formule matematiche.
 - I linguaggi di composizione TeX e LaTeX.
 - Il linguaggio grafico MetaPost (cenni).
- Approfondimenti:
 - La dimostrazione di un teorema.
 - L'esposizione di un algoritmo.
 - La documentazione di un (frammento di) programma.

Testi consigliati:

1. Tobias Oetiker, Hubert Partl, Irene Hyna e Elisabeth Schlegl. *Una (mica tanto) breve introduzione a LaTeX 2e*. Disponibile in formato elettronico.
2. Donald E. Knuth, Tracy L. Larrabee, Paul M. Roberts. *Mathematical Writing*, Mathematical Association of America, 1989. ISBN 0-88385-063-X.
3. Donald E. Knuth, *The TeXbook*, Addison-Wesley, 1984. ISBN 0-201-13448-9.
4. Leslie Lamport. *LaTeX: A Documentation Preparation System*, 2nd edition, Addison-Wesley, 1994. ISBN: 0-201-52983-1.
5. Helmut Kopka, Patrick W. Daly. *A Guide to LaTeX2e*, 2nd edition, Addison-Wesley, 1995. ISBN: 0-201-42777-X.
6. Michel Goossens, Frank Mittelbach, Alexander Samarin. *The LaTeX Companion*, Addison Wesley, 1994. ISBN: 0-201-54199-8.

SISTEMI INFORMATIVI GEOGRAFICI

Docente: A. Clerici

Crediti (CFU): 3

Finalità: scopo del corso è quello di introdurre gli studenti all'utilizzo dei Sistemi Informativi Geografici (GIS). Dopo un esame delle caratteristiche e delle modalità operative dei GIS in generale, vengono analizzate le caratteristiche specifiche di GRASS (Geographic Resources Analysis Support System), un GIS di pubblico dominio particolarmente efficiente ed idoneo ad un utilizzo didattico.

In base alla considerazione che il miglior apprendimento di un GIS avviene mediante il suo utilizzo, il corso si svolge quasi esclusivamente attraverso l'esecuzione di esercizi che riguardano dapprima i comandi di più frequente utilizzo e successivamente le operazioni di acquisizione dei dati e la rappresentazione tridimensionale dei risultati, argomento quest'ultimo di particolare utilità nelle applicazioni geologiche e geomorfologiche.

A conclusione del corso vengono illustrati alcuni esempi di utilizzo avanzato di GRASS per evidenziare le notevoli potenzialità del sistema.

Programma del corso:

- Caratteristiche dei GIS.
- Caratteristiche del GIS GRASS (Geographic Resources Analysis Support System).
- I principali comandi di GRASS, con esercizi introduttivi.
- L'acquisizione dei dati tramite GRASS, con esercizi.
- L'utilizzo di NVIZ per la rappresentazione tridimensionale dei dati, con esercizi.
- Esempi di applicazioni avanzate di GRASS.

Testi consigliati:

1. A. Clerici. *GRASS: esercizi introduttivi*, (GRASS5.0, ed. 1/2003).
2. A. Clerici. *Manuale pratico per l'uso di v.digit* (GRASS5.0: GRASS-DIGIT Modified 4.10).
3. A. Clerici. *Manuale pratico per l'uso di nviz* (GRASS5.0).

SISTEMI OPERATIVI (+ LABORATORIO)

Docente: M. Pellegrini

Docente per il laboratorio: R. Alfieri

Crediti (CFU): 6 per la parte teorica, 4 per il laboratorio

Finalità: il corso introduce i principi ed i concetti fondamentali su cui si basano i sistemi operativi. In particolare, vengono analizzate le tecniche che consentono di coordinare e gestire le risorse di un sistema di elaborazione e che permettono di trasformare la macchina fisica in una macchina astratta, dotata di funzionalità più convenienti per l'utente.

Programma del corso:

- *Introduzione.*
Richiami sulla struttura del calcolatore
- *Processi e Thread.*
Modello a processi, stato, generazione e terminazione. Il modello a thread, uso e implementazione. Paradigmi di interazione: condivisione, sincronizzazione, comunicazione. Scheduling di thread e processi.
- *Le Risorse.*
Modelli di gestione delle risorse. Politiche elementari di gestione. Il problema dello stallo: caratterizzazione, metodi per evitarlo, impedirlo, riconoscerlo ed eliminarlo. Attesa indefinita. Alcuni problemi classici.
- *Gestione della memoria.*
Il problema di base. Swapping. Paginazione. Algoritmi di sostituzione. Cenni sulla tecnica di segmentazione ed sulle problematiche implementative.
- *Gestione dei dispositivi di Ingresso/Uscita.*
Dispositivi e processi controller. Caratteristiche del software di I/O, vari livelli di gestione: interruzioni, driver dei dispositivi, I/O software indipendente dai dispositivi. Gestione di alcuni dispositivi: dispositivi a caratteri, dischi magnetici.
- *File System.*
Modello logico del sistema di archiviazione, file, directory. Implementazione di un file system. Esempi di file system.
- *Cenni a problematiche avanzate.*
Approfondimento di un problema nell'ambito sistemi operativi.

Programma per la parte di Laboratorio:

- Il sistema operativo Unix/Linux. Storia, struttura, interfaccia utente, installazione e amministrazione.
- La shell di Unix. La shell Bash, shell scripting, i filtri.
- Programmazione di Sistema in C. Chiamate e librerie di sistema, controllo dei processi, comunicazioni tra processi, thread e multithreading, accesso al file-system e I/O.
- Programmazione di Sistema in JAVA. Introduzione, sintassi del linguaggio, OOP, eccezioni, Thread, Applet.

Testi consigliati:

1. Andrew S. Tanenbaum. *I Moderni Sistemi Operativi*, Seconda Edizione, Jackson Libri (Aprile 2002). ISBN: 8825618980 (versione italiana)
2. Andrew S. Tanenbaum. *Modern Operating Systems*, Second Edition, Prentice Hall 2001 ISBN 0-13-031358-0 (versione inglese)

STRUMENTI PER APPLICAZIONI WEB

Docente: E. Calabrese

Crediti (CFU): 5

Finalità: il corso si prefigge di presentare i metodi e gli strumenti per costruire applicazioni Web, di fornire una panoramica delle tecnologie disponibili e di mettere lo studente in grado di sviluppare applicazioni Web interattive utilizzando alcune delle principali tecnologie. Esercitazioni: le esercitazioni costituiscono il 40% circa (2 CFU) del corso. Durante le esercitazioni verranno studiate delle applicazioni Web preesistenti (iscrizione agli esami, test on line, database del personale, ecc.) e ne verranno sviluppate alcune che comportano l'interfacciamento con basi di dati. Propedeuticità consigliate: basi di dati (anche in concomitanza).

Programma del corso:

- Introduzione a Internet e al Web
- Tecnologie lato client
 - HTML
 - Cascading Style Sheets
 - Javascript
 - Applet Java
 - Introduzione all'XML
- Tecnologie lato server
 - I server Web
 - Programmazione CGI e cookies
 - Perl
 - Cenni a Java e ai Servlet
 - Introduzione al PHP
 - Accesso ai database via Web
 - Cenni a usabilità, accessibilità e sicurezza
- Servizi Web universitari

Testi consigliati:

1. S. Guelich, S. Gundavaran, G. Birznieks. *CGI Programming with Perl*, O'Reilly, 2000.
2. R. W. Sebesta. *Programmare il World Wide Web*, McGraw-Hill, 2003; (si consiglia la seconda edizione, in lingua inglese: *Programming the World Wide Web (2nd Edition)*, Addison-Wesley, 2002).
3. Deitel, Deitel, Nieto, McPhie. *Perl - How to Program*, Prentice Hall, 2001.

TEORIA DEI NUMERI E CRITTOGRAFIA

Docente: A. Zaccagnini

Crediti (CFU): 4

Programma del corso:

1. Richiami alla teoria dei gruppi e dei campi finiti
 - Teoremi di Fermat, Eulero e Wilson, struttura dell'anello $\mathbb{Z}/p\mathbb{Z}$.
 - Teorema di Gauss: esistenza delle radici primitive (generatori) dei gruppi $(\mathbb{Z}/p\mathbb{Z})^*$, p primo.
 - Condizioni necessarie e sufficienti per la primalità. Pseudoprimi di Fermat, di Eulero, pseudoprimi forti.
 - Cenni al Teorema di Agrawal, Kayal, Saxena.
2. Algoritmi fondamentali
 - Algoritmo di Euclide, crivello di Eratostene, criteri di primalità.
 - Algoritmi di fattorizzazione esponenziali: divisione per tentativi, metodo di Lehman, metodo *rho* di Pollard, metodo $p - 1$ di Pollard.
 - Algoritmi di fattorizzazione subesponenziali: crivello quadratico.
 - Algoritmo di Gauss per la determinazione delle radici primitive.
 - Logaritmo discreto: algoritmo di Shanks.
3. Applicazioni alla crittografia
 - Cenni alla crittografia classica.
 - Crittografia a chiave pubblica: Diffie-Hellman, RSA, Massey-Omura, ElGamal, Rabin.
 - Firma digitale.
 - Protocolli crittografici (cenni).

Testi consigliati:

1. R. Crandall, C. Pomerance. *Prime numbers. A computational perspective*, Springer, New York, 2001.
2. G. H. Hardy & E. M. Wright. *An Introduction to the Theory of Numbers*, quinta edizione, Oxford Science Publications, Oxford, 1979.
3. N. Koblitz. *A Course in Number Theory and Cryptography*, seconda edizione, Springer, 1994.
4. A. Languasco, A. Zaccagnini. *Introduzione alla Crittografia Moderna*, in preparazione, 2004.

TEORIA DEI SEGNALI

Docente: G. Ferrari

Crediti (CFU): 5

Finalità: il corso si propone di fornire allo studente una conoscenza di base della teoria della probabilità e variabili aleatorie, con applicazioni all'ingegneria.

Programma del corso:

- Teoria della probabilità: richiami di teoria degli insiemi; assiomi di teoria della probabilità e conseguenze. Elementi di calcolo combinatorio. Probabilità condizionata, teorema della probabilità totale e formula di Bayes. Prove ripetute.
- Variabili Aleatorie: introduzione al concetto di funzione di densità di probabilità. Definizione formale della funzione densità di probabilità e della sua primitiva, cioè la funzione cumulativa di distribuzione. Delta di Dirac. Variabili aleatorie continue e discrete.
- Trasformazioni di variabili aleatorie: trasformazione di una singola variabile aleatoria e teorema fondamentale. Valor medio e teorema dell'aspettazione. Momenti e funzione generatrice dei momenti. Formula di Bayes mista e versione continua del teorema della probabilità totale. Coppie di variabili aleatorie e trasformazioni di coppie di variabili aleatorie. Estensioni a sistemi di n variabili aleatorie. Teorema dell'aspettazione e della media condizionata per n variabili aleatorie. Correlazione. Indipendenza e incorrelazione.
- Legge dei grandi numeri e sua interpretazione statistica. Interpretazione statistica di covarianza. Coefficiente di correlazione. Teorema del limite centrale. Teorema di De-Moivre Laplace.

Testi consigliati:

1. A. Bononi. *Appunti di Teoria dei Segnali*, (dispense).
2. G. Prati. *Esercizi di teoria delle variabili casuali*, (raccolta esercizi svolti).

TEORIA DELL'INFORMAZIONE

Docente: G. Ricci

Crediti (CFU): 6

Programma del corso:

- Compressione dati: entropia in spazii prove finiti, omomorfismi di catenazione, compressione a memoria limitata, (dis-)uguaglianze di Kraft, teoremi Fano e Shannon, esempi compressione a memoria illimitata.
- Introduzione trasmissione informazione: spazio messaggi, cilindri, entropia, stazionarietà, canale, cenno teoremi Von Neumann-Shannon, esempi codici diagnostici ed autocorrettivi.
- Cenni teoria dell'Informazione Algoritmica: generazione con macchine Turing, quantità informative algoritmiche, (dis-)uguaglianze asintotiche, rappresentazioni numeri naturali, casualità algoritmica e teorema del minorente.

Testi consigliati:

1. S. Guiasu. *Information Theory with applications*, McGraw-Hill, 1977.
2. S. Guiasu, R. Theodorescu. *La théorie mathématique de l'information*, Dunod, Paris 1968.
3. G. J. Chaitin. *Information, Randomness and Incompleteness*, World Scientific, 1987.
4. F. Fabris. *Teoria dell'Informazione, codici, cifrari*, Bollati Boringhieri, 2001.