

Modifiche syllabi Corsi Laurea Triennale (tabella 1)

1 anno laurea triennale

Algoritmica e Laboratorio

nel paragrafo:

Algoritmi per Grafi: rappresentazione, algoritmi di visita, Albero di Copertura Minimo, Cammini Minimi (Dijkstra), Componenti (fortemente) connesse.

eliminare

Componenti(fortemente) connesse

Fisica

Prerequisiti

Conoscenze matematiche di base relative alla geometria euclidea, alla geometria analitica ed alla trigonometria. Conoscenza del concetto generale di funzione matematica, derivata ed integrale.

Obiettivi di apprendimento

Introduzione al linguaggio della fisica come descrizione matematica dei fenomeni naturali. Conoscenza delle leggi fondamentali della meccanica, dell'elettricità e del magnetismo con particolare attenzione all'uso del concetto di conservazione delle grandezze fisiche.

Descrizione di problemi fisici attraverso semplici modelli matematici.

Capacità di costruzione di modelli deterministici in grado di descrivere e predire l'evoluzione di sistemi naturali.

Conoscenze: Conoscenze di base. *Capacità:* Capacità di apprendimento, di comprensione e di verifica di un modello matematico della realtà.

Descrizione

Il corso è suddiviso in due parti.

La prima parte, che impiega circa 36 ore delle 48 previste, fornisce un'introduzione sistematica ai principi della fisica classica. La verifica della comprensione dei concetti si basa su semplici applicazioni che richiedono semplici conoscenze di algebra, geometria ed occasionalmente di analisi matematica. Gli argomenti affrontati nella prima parte del corso sono: la descrizione di alcune interazioni fondamentali in natura (interazione gravitazionale ed elettrica), le leggi del moto (derivate dai principi di Newton), i concetti di lavoro ed energia, i principi di conservazione di energia e quantità di moto; il campo magnetico, sua origine e suo effetto su cariche in moto; i circuiti a corrente continua ed alternata con cenni ad applicazioni per l'elettronica.

La seconda parte del corso è dedicata allo studio di alcuni fenomeni "naturali" (evoluzione di sistemi meccanici, elettrici, ecologici, economici, termodinamici). Si discutono le possibilità e i limiti di una loro descrizione mediante equazioni differenziali ordinarie. Della poca teoria matematica necessaria e non nota da analisi matematica si danno i risultati in forma direttamente utilizzabile per la discussione dei sistemi studiati. Simulazioni numeriche vengono utilizzate per studiare i sistemi complessi.

Programma

Grandezze fisiche e operazioni di misura. Campioni, unità di misura. Grandezze vettoriali e operazioni fra vettori. Cinematica del punto materiale: velocità e accelerazione. Moto circolare.

Forze a distanza e di contatto. Esempi di forze: forza gravitazionale, forza elettrostatica. La legge di gravitazione universale. Forza di gravità vicino alla superficie terrestre. Concetto di campo di forza. Legge di Gauss con particolare enfasi sul principio di sovrapposizione e sul concetto di simmetria. Dinamica: le leggi del moto (derivate dai principi di Newton), con forze costanti e non. Esempi: forze proporzionali alla velocità, forze dipendenti dalla posizione (forze elastiche e legge di Hooke).

Forze di contatto. Vincoli e reazioni. Forze esplicate dai vincoli. Forze di attrito statico e di attrito dinamico. Lavoro ed energia, forze conservative, energia cinetica e potenziale, i principi di conservazione di energia e quantità di moto. Sistemi descrivibili con equazioni differenziali lineari a coefficienti costanti omogenee e non omogenee. Esempi: moto in fluido viscoso (primo ordine), oscillatore armonico e pendolo semplice in regime delle piccole oscillazioni (secondo ordine).

Conduttori all'equilibrio: capacità ed energia elettrostatica. Corrente elettrica, conduzione nei metalli. Legge di Ohm. Potenza dissipata dalla corrente e Legge di Joule. Resistenze in serie e in parallelo. Reti semplici di condensatori e resistenze. Scarica e carica di un condensatore attraverso una resistenza. Magneti permanenti in natura. Dipolo magnetico. Linee di campo magnetico. Forza di Lorentz. Forza magnetica agente su un elemento di corrente. Campo generato da un elemento di corrente e da varie configurazioni. Legge di Biot-Savart. Circuitazione e Teorema di Ampère. Definizione di flusso del campo magnetico concatenato con un circuito. Induzione elettromagnetica: legge di Faraday-Lenz. Spira rotante in campo magnetico. Autoinduzione. Risoluzione di semplici circuiti a corrente alternata con cenni ad applicazioni per l'elettronica come filtri, circuiti risonanti RLC, linee di trasmissione.

Sistemi deterministici lineari del primo e secondo ordine. Stabilità. Esempi fisici: oscillatore armonico ideale, smorzato, instabile. Operatore di evoluzione, autovalori e autovettori dell'operatore di evoluzione e loro significato fisico. Sistemi lineari e spazio delle fasi. Alcuni sistemi non lineari. Vari esempi: equilibri multipli, stabilità lineare, cicli di isteresi, cicli limite.

Bibliografia

“Gettys Fisica 1 – Meccanica” a cura di Gianni Vannini, 2011 McGraw-Hill

“Gettys Fisica 2 - Elettromagnetismo – Onde” a cura di Giovanni Cantatore, Lorenzo Vitale, 2011 McGraw-Hill

Per la parte del corso relativa all'evoluzione di sistemi deterministici si consigliano i primi due capitoli degli appunti del corso di “Modelli della Fisica” di F. Cornolti, disponibili on-line.

Indicazioni metodologiche

Il processo di apprendimento è organizzato in sequenza logica con valutazione intermedia degli obiettivi di apprendimento raggiunti. Si accompagna l'introduzione dei concetti fondamentali con esempi pratici durante le esercitazioni in aula. Viene inoltre fornito on-line materiale opzionale per lo studio a casa (come esercizi svolti, tutorials, appunti delle lezioni, dispense per approfondimento su singoli argomenti, etc.).

Modalità di esame

Scritto e orale. All'orale si accede se lo scritto è superato con una votazione superiore o pari a 15/30. Dall'esame scritto è esonerato chi supera le prove in itinere con una media superiore o pari a 15/30. Le prove in itinere saranno due con la possibilità di recupero di una sola delle due prove a fine corso. Modalità degli scritti è la soluzione con svolgimento esteso di problemi di fisica analoghi a quelli trattati durante il corso. L'orale verte su domande relative agli argomenti illustrati nel corso e loro semplici applicazioni.

Come complemento facoltativo, gli studenti che lo vogliono possono selezionare un progetto relativo all'evoluzione di sistemi deterministici, scelto fra gli argomenti trattati nel dettaglio a lezione. Lo studente può implementare il progetto e redigere una breve relazione di poche pagine che illustra i concetti fisici rilevanti al problema, il metodo usato, le soluzioni numeriche impiegate. Questa attività facoltativa, che può essere vista come una specie di esercitazione in classe prolungata con un lavoro di qualche ora individuale, contribuisce alla valutazione in sede di orale (senza punteggio indipendente).

2 anno laurea triennale

Sistemi Operativi

sostituire

Funzioni, struttura e componenti dei Sistemi Operativi
- Processi e thread: concetti, realizzazione e gestione

- Sincronizzazione e comunicazione tra processi e tra thread
- Gestione del processore: obiettivi e politiche
- Errori di interazione: il problema dello stallo
- Gestione della memoria; memoria virtuale
- Gestione dei dispositivi di I/O e di memoria di massa
- Sistema di archiviazione e gestione degli archivi
- Protezione e sicurezza
- Introduzione ai sistemi distribuiti
- Studio di casi: i sistemi della famiglia Unix/Linux e Windows NT

con

- Funzioni, struttura e componenti dei Sistemi Operativi
- Nucleo e Processi
- Gestione dei processi e del processore: concorrenza e sincronizzazione tra processi e tra thread, scheduling
- Gestione della memoria, memoria virtuale e cache
- Gestione dei dispositivi di I/O e di memoria di massa
- Sistema di archiviazione e gestione degli archivi

Architettura degli Elaboratori (Vanneschi)
 sostituire architetture con architettura

3 anno triennale

Ingegneria del Software

sostituire:

Metodologie e strumenti per la progettazione, realizzazione, verifica, validazione e misurazione di sistemi software.

Contenuti

Processo di sviluppo software: problemi della produzione del software, modelli di ciclo di vita.

Analisi del dominio: modelli statici (classi e associazioni) e dinamici (attività, macchine a stati).

Analisi dei requisiti: modello statico (casi d'uso) e dinamici (narrative, diagrammi di robustezza).

Progettazione architettonica: modelli statici (viste strutturali e logistiche) e dinamici (vista componenti/connettori).

Progettazione di dettaglio: modello statico delle componenti (strutture composite) e modello dinamico (interazioni).

Verifiche e prove: obiettivi e pianificazione delle verifiche, progettazione e valutazione delle prove.

con

“Il corso introduce i problemi della produzione di sistemi software e illustra alcuni processi di sviluppo software. Viene presentato in dettaglio un processo di sviluppo guidato dai modelli, definendo modelli statici e modelli dinamici per le principali fasi del processo: analisi del dominio, analisi dei requisiti, progettazione architettonica e dettaglio. Infine si affronta il tema della verifica: obiettivi e pianificazione delle verifiche, progettazione e valutazione delle prove.

Reti di Elaboratori e Laboratorio, modulo di programmazione di rete

Sostituire

- JAVA: Il Modello di Concorrenza, Threads, Risorse Condivise, Cooperazione di Threads, Thread Pooling

- Sockets TCP, Indirizzi IP, Streams, URL, Connessioni HTTP
- Sockets UDP, Multicast
- Oggetti Distribuiti, Callbacks
- Strumenti per l'Esecuzione di Programmi Distribuiti

con

il corso si propone di introdurre i concetti fondamentali della programmazione concorrente in JAVA e di fornire gli strumenti per lo sviluppo di applicazioni di rete in JAVA. In particolare nella prima parte del corso viene introdotto il modello di concorrenza di JAVA (threads, mutua esclusione, cooperazione di Threads, Thread Pooling). Nella seconda parte del corso vengono introdotti i socket sia TCP che UDP, riprendendo il concetto di stream e approfondendo il concetto di serializzazione di oggetti. Nella terza parte del corso viene introdotta la programmazione ad oggetti distribuita, con particolare riferimento al meccanismo di Remote Method Invocation. Infine l'ultima parte del corso prevede un'attività pratica su una rete virtuale, in particolare si forniscono gli strumenti per la configurazione di una rete: interfacce, routing, DNS. Il corso si conclude con l'assegnazione di un progetto didattico.

Laurea Triennale: Complementari

Logistica

sostituire

Il corso si propone di presentare la struttura ed il funzionamento dei sistemi logistici ed alcune delle principali metodologie per la loro gestione ottimale, con particolare enfasi sui modelli di programmazione a numeri interi. Verranno inoltre discusse alcune delle principali tecniche algoritmiche necessarie a risolvere i problemi nelle dimensioni richieste dalle applicazioni reali ed agli studenti verrà fornita la capacità di utilizzare in modo appropriato alcuni dei software disponibili che implementano tali metodologie.

con

Il corso presenta la struttura ed il funzionamento dei sistemi logistici, con enfasi su aspetti di modellazione. Vengono descritti alcuni rilevanti problemi di progetto di reti logistiche (problemi di localizzazione e problemi di trasporto), e problemi decisionali di tipo tattico. I problemi decisionali presentati vengono formulati mediante modelli matematici (Programmazione Lineare e Programmazione Lineare Intera), alcuni dei quali sono implementati, risolti ed analizzati mediante l'ausilio di strumenti software adeguati.

Simulazione

Sostituire

Il corso si propone di fornire allo/a studente/ssa gli strumenti di base per la costruzione e l'utilizzo di modelli di simulazione. Un rilevante spazio verrà dato alle applicazioni. Gli argomenti trattati verranno sviluppati anche attraverso piccoli progetti da svolgere durante l'anno: tali progetti costituiranno una componente essenziale nella valutazione conclusiva. Introduzione ai modelli di simulazione Simulazione discreta Il sistema da modellare Approcci alla modellazione Simulazione per processi e per attività Funzioni di distribuzione e test statistici Variabili casuali Distribuzioni discrete e continue Stima di parametri Test di ipotesi Analisi e scelta dei dati di input Distribuzioni empiriche Analisi dei dati di input Numeri pseudocasuali Analisi dei dati di output Analisi del transitorio Tecniche per la riduzione della varianza Modelli di Dinamica dei Sistemi Diagrammi causali Livelli e flussi Ritardi Exponential smoothing Scelta dell'unità di tempo e metodi di integrazione

Con

Il corso si propone di fornire gli strumenti di base per la costruzione e l'utilizzo di modelli di simulazione. Dopo un'introduzione alla simulazione saranno presentate le principali tecniche di simulazione, con particolare attenzione alla simulazione discreta. Una parte consistente del corso sarà dedicata alle metodologie statistiche per l'analisi dei dati di input e di output. Nell'ultima parte del corso saranno presentati i principali concetti della System Dynamics e della simulazione per agenti autonomi.

Laboratorio di Applicazioni Internet

Sostituire

L'evoluzione delle architetture software: dai mainframe al client/server a multitier/Internet. L'impatto dei Firewall sulla progettazione delle applicazioni Internet
Introduzione alle tecnologie di base
Livello Applicativo (Web Server)
Livello Applicativo (Back-End)
Le problematiche di Sicurezza

Con

Il Corso presenta le architetture, i protocolli e gli strumenti di sviluppo per la realizzazione di Applicazioni Internet, permettendo agli studenti di esercitarsi nell'uso degli stessi nello sviluppo incrementale di un articolato progetto didattico in ambiente Java Enterprise (JEE).

Syllabus

- Architettura delle Applicazioni Internet
- Introduzione al Protocollo HTTP, uso applicativo lato client e server
- Introduzione all'XML, tecniche di parsing e validazione
- Introduzione a SOAP/WSDL per la realizzazione di applicazioni di tipo Web Services
- Uso evoluto dei Web Services: realizzazione di nodi intermedi, uso dei servizi infrastrutturali (WS-*)
- La gestione delle transazioni nelle Applicazioni Internet
- Aspetti di Sicurezza nelle Applicazioni Internet

Macchine virtuali per linguaggi di programmazione di alto livello

Prerequisiti

Programmazione con paradigmi imperativo e object oriented. Discreta conoscenza del linguaggio Java. Conoscenza di base delle architetture dei calcolatori e dei sistemi operativi

Descrizione

Una macchina virtuale è un insieme di programmi (software) che simulano il comportamento di una macchina diversa dalla macchina fisica su cui il software viene eseguito. I vantaggi della virtualizzazione possono essere, a seconda del contesto: l'indipendenza dalla piattaforma, la portabilità, la robustezza, affidabilità e la protezione del software. Nella parte introduttiva del corso, si esaminano gli ambiti e le tipologie di macchine virtuali focalizzandosi successivamente sulle high level language virtual machines (HLL VMs). Verranno considerate la Pascal P-code VM, the JVM e la CLI VM, il corso approfondisce lo studio delle caratteristiche e dell'implementazione della JVM, e comprende una parte pratica, in laboratorio, per la generazione di programmi in byte code utilizzando l'assembler Oolong

English Description

Requirements: Programming with imperative and object-oriented paradigms. Good knowledge of the Java language. Basic knowledge of computer architectures and operating systems.

Description: A virtual machine is a software that simulates the behavior of a machine other than the physical machine on which the software is running. The benefits of virtualization can be, depending on the context: platform independence, portability, robustness, reliability and security of software. In the introductory part of the course, we will examine different types of virtual machines and successively we will focus on high level language virtual machines (VMs HLL). The Pascal P-code VM, the VM JVM and the CLI, will be considered. The course focuses on the JVM studying its characteristics and implementation and includes a practical part, in the laboratory, for byte code programming, using the 'Oolong assembler for translation into byte code.

Testi di consultazione

- Virtual Machines, Versatile Platforms for System and Processes, J.E.Smith and R. Nair, Morgan Kaufmann Pub. Elsevier,2005, ISBN-13: 978-1-55860-910-5, ISBN-10:55860-910-5.
- Programming for the JVM, J. Engel, Addison Wesley, 1999, ISBN 0-201-30972-6.
- Java Virtual Machine specifications, T. Lindholm and F. Yellin, Addison Wesley, 1996, ISBN0-201-63452-X

Modalità di esame: Progetto e orale