

# **Ing. Informatica**

**Disciplina:** 0065206 **BASI DI DATI**

ING-INF/05

**Corso di Studio:** IIN

**Crediti:** 6 **Tipo:** A

**Note:**

**Docente:** PALA PIETRO

P2 ING-INF/05

**Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

Sistemi Informativi e Basi di Dati; Il modello relazionale (definizione di relazione, vincoli di integrità); Algebra relazionale (unione, intersezione, differenza, selezione, proiezione e join); Il linguaggio SQL (tipi di dati, creazione tabelle, specifica vincoli di integrità, il comando select, clausole di raggruppamento, ordinamento, operatori aggregati, interrogazioni nidificate, viste, aggiornabilità delle viste, viste ricorsive); Progetto di basi di dati: Il modello Entity-Relationship (entità, relazioni, attributi, cardinalità, generalizzazioni); Progettazione logica (analisi delle ridondanze, ristrutturazione di schemi ER, traduzione nel modello relazionale); Forme normali (dipendenze funzionali, decomposizioni, copertura minimale, forme normali prima, seconda, terza e Boyce Codd, procedure di normalizzazione); Basi di dati attive: Triggers; Transazioni;

**Disciplina:** 0065058 **FONDAMENTI DI AUTOMATICA**

ING-INF/04

**Corso di Studio:** IIN IEL

**Crediti:** 6 **Tipo:** A

**Note:**

**Docente:** CHISCI LUIGI

P1 ING-INF/04

**Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

### 1. INTRODUZIONE

- Problemi di controllo
- Sistemi di controllo
- Richiami sui sistemi dinamici lineari tempo-invarianti (LTI)
- Poli/zeri e comportamento dinamico.

### 2. ANALISI DI SISTEMI A RETROAZIONE

- Sistemi interconnessi (connessioni serie, parallelo e retroazione)
- Stabilità interna
- Diagramma di Nyquist
- Criterio di Nyquist
- Margini di stabilità
- Il luogo delle radici
- Specifiche statiche: errori a regime e tipo del sistema; principio del modello interno.
- Specifiche dinamiche nel dominio del tempo (sovraelongazione, tempo di assestamento e tempo di salita)
- Specifiche dinamiche nel dominio della frequenza (picco di risonanza, banda passante, margine di fase e pulsazione di attraversamento)
- Moderazione della variabile di controllo
- Correlazioni empiriche fra le varie specifiche

### 3. SINTESI DI SISTEMI DI CONTROLLO A RETROAZIONE

- Funzioni compensatrici elementari: funzione attenuatrice ed anticipatrice
- Sintesi per tentativi nel dominio della frequenza
- Sintesi con il luogo delle radici
- Regolatori PID (argomento non svolto)
- Sintesi diretta (argomento non svolto)
- Cenni sui sistemi di controllo digitale (argomento non svolto)

**Disciplina:** 0065169 **INFORMATICA INDUSTRIALE**

ING-INF/05

**Corso di Studio:** IIN AUS ELM IDT ELS

**Crediti:** 6 **Tipo:** A

**Note:** IDT ELS NON ATTIVO

**Docente:** FANTECHI ALESSANDRO

P1 ING-INF/05

**Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

- 1) Caratteristiche generali dei sistemi embedded
- 2) Richiami su macchine a stati finiti, automi a stati finiti riconoscitori di linguaggi – classificazione di chomsky
- 3) Implementazione di macchine a stati
- 4) Realizzazione sistemi event-driven con interruzioni
- 5) Tipici cicli di controllo - vincoli real-time – wcet - task periodici
- 6) Real time Scheduling
- 7) Fixed priority scheduling – rate monotonic priority assignment – Earliest deadline first
- 8) Priority inversion - Priority inheritance
- 9) Sistemi operativi real-time
- 10) Introduzione ai processori di utilizzo industriale - classificazione in MPU, MCU, DSP, PLC, PC industriali
- 11) Dependability – concetti e terminologia
- 12) Valutazione dell’Affidabilità – failure rate, MTTF
- 13) Modello MIL-HDBK 217 F per valutazione Affidabilità di componenti Hw
- 14) Modello combinatorio per la valutazione dell’affidabilità Modelli serie, parallelo, N su M
- 15) Modello markoviano per la valutazione degli attributi di dependability
- 16) Disponibilità, Manutenibilità, MTBF, MTTR
- 17) Safety
- 18) Tecniche di valutazione qualitativa affidabilità: FMEA / HAZOP / FTA
- 19) Meccanismi di rilevazione degli errori - duplicazione e confronto
- 20) Codici rilevatori di errore
- 21) Principi dei codici correttori di errore e loro applicazioni
- 22) Fault masking TMR NMR
- 23) Architetture a ridondanza ibrida
- 24) Riconfigurabilità - hot spare/cold spare
- 25) Ridondanza per diversità - Software fault tolerance
- 26) Esempi di sistemi dependable
- 27) Introduzione ai metodi formali per lo sviluppo, la specifica e la verifica del SW
- 28) Introduzione alla verifica formale - Model checking
- 29) La certificazione software e la Normativa CENELEC
- 30) Testing del software

**Note:**

1. Introduzione. I problemi dell'ingegneria del software. Il software come prodotto industriale. Il ciclo di vita del software. Cenni sui differenti modelli di cicli di vita. Principi generali di programmazione: Astrazione; struttura dei programmi; modularizzazione. incapsulamento dell'informazione; tipi di dati astratti.

2. Programmazione orientata agli oggetti. Classi, oggetti, ereditarietà. Principi e metodi di programmazione OO. Relazioni tra oggetti: associazione, aggregazione, derivazione/generalizzazione, composizione di oggetti. Delega. Polimorfismo. Il modello MVC. Esempi di Design Patterns.

3. UML. Il linguaggio UML. Generalità. Il diagramma delle classi: classi astratte interfacce. Associazioni, dipendenze e altre relazioni tra classi. Il diagramma dei casi d'uso. Diagrammi di sequenza, di collaborazione, di attività e di stato. Packaging e deployment.

4. Analisi - Progettazione - Sviluppo. Stesura dei requisiti in forma testuale (Std IEEE 830); costruzione del modello di dominio. Analisi dei casi d'uso: diagramma dei casi d'uso, esame dettagliato dei singoli casi d'uso. Analisi di robustezza: identificazioni delle funzionalità applicative, riesame dei casi d'uso eliminazione di possibili ambiguità residue, riesame del modello di dominio. Realizzazione dei casi d'uso: impiego dei diagrammi di sequenza. Individuazione delle responsabilità delle classi; definizione delle relative interfacce. Progetto di dettaglio, transizione verso lo sviluppo. definizione dei casi di test dai casi d'uso. Sviluppo guidato dal test (TDD). (Nel presentare questa parte si farà normalmente riferimento a casi concreti.). Cenni al metodo dell'analisi del Flusso Dati.

5. Miscellanea Tipologie applicative. Sistemi "Enterprise". Persistenza. Programmazione OO e basi di dati relazionali. ORM. Pattern architetturali. Cenni su tool e framework per lo sviluppo di sistemi software. Applicazioni a prevalente aspetto di controllo. Cenni sui sistemi di tempo reale. I ciclo di vita: il modello a cascata e le sue fasi, il modello UP, modelli agili, il modello XP. Standard industriali di riferimento. Impatto degli standard industriali sul ciclo di vita: documentazione, pianificazione, analisi del rischio. Misurare il software. Metriche software (SLOC, Function Points). Previsione di costi e tempi di sviluppo (COCOMO). Qualità del software.

**Disciplina:** 0065204 **INTELLIGENZA ARTIFICIALE**

ING-INF/05

**Corso di Studio:** IIN AUS

**Crediti:** 6 **Tipo:** A

**Note:**

**Docente:** SODA GIOVANNI

P1 ING-INF/05

**Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

Argomenti:

Tecniche di ricerca:

Spazi di ricerca- Ricerca cieca- Depth-first - Breadth-first – Iterative Deepening-Ricerca euristica: Hill Climbing - Best- first - Simulated annealing - Algoritmo A\*- IDA\*

Rappresentazione della conoscenza e reasoning:

La logica dei predicati - Tecniche di inferenza nella logica dei predicati- Modus Ponens- Dimostrazione automatica dei teoremi - Risoluzione - Algoritmo di unificazione - Cenni sulla programmazione logica.

Apprendimento:

Concetti introduttivi - Apprendimento induttivo - Spazio delle Versioni- Alberi di decisione - Apprendimento con supervisione nelle reti neurali - Disamina delle principali architetture - Overfitting - Esempi applicativi

**Disciplina:** A000116 **MATEMATICA DISCRETA - Modulo ALGEBRA** MAT/03

**Corso di Studio:** IIN **Crediti:** 3 **Tipo:** A

**Note:** Modulo del corso Matematica Discreta 6 CFU

**Docente:** BATTAGLIA FIAMMETTA RC MAT/03 **Copertura:** AFF03

**Ente appartenenza:** Dip. Matematica Applicata "G.Sansone"

---

Vedi anche il registro delle lezioni alla pagina:

a.a. 2008/09

<http://www.dma.unifi.it/~fiamma/registrodiscreta0809.html>

a.a. 2009/10

<http://www.dma.unifi.it/~fiamma/registrodiscreta0910.html>

Programma dettagliato:

I numeri interi:

--definizione di gruppo, anello, campo.

L'insieme  $\mathbb{Z}$  dei numeri interi è un gruppo abeliano rispetto alla somma, e' un anello commutativo con le operazioni di somma e prodotto.

Divisione con resto in  $\mathbb{Z}$ . Congruenze.

L'algoritmo dei quadrati ripetuti.

Massimo comun divisore: esistenza e unicità'.

Algoritmo Euclideo per determinare il massimo comun divisore,  $\gcd(m,n)$ , tra due interi  $m$  e  $n$ .

Algoritmo Euclideo esteso per determinare interi  $x$  e  $y$  tali che  $xm+yn=\gcd(m,n)$ .

La congruenza è una relazione di equivalenza.

Teorema cinese dei resti

Definizione della funzione  $\phi$ ; di Eulero sui naturali.

Teorema di Eulero.

Numeri primi. Calcolo della funzione di Eulero di un numero nota la fattorizzazione in primi.

Il sistema RSA.

Anello delle classi di resto modulo  $n$  ( $\mathbb{Z}/n\mathbb{Z}$ ).

$\mathbb{Z}/n\mathbb{Z}$  è un campo se e solo se  $n$  è primo, il campo  $F_n$ .

Polinomi.

Anello dei polinomi a coefficienti in un anello, in un campo. Grado.

Algoritmo della divisione. Esempi.

Radice di un polinomio.

Molteplicità'.

Sia  $f$  un polinomio a coefficienti in un campo  $K$ , con radici  $a_1, \dots, a_r$ . Allora  $f=q(x-a_1)^{e_1} \dots (x-a_r)^{e_r}$ , dove  $q$  è un polinomio a coefficienti nel campo  $K$ , senza radici in  $K$ .

Derivata  $Df$  di un polinomio  $f$  e sue proprietà'.

Radici dell'unità' nel campo complesso. Radici primitive. Polinomi ciclotomici e loro proprietà'.

Caratteristica di un campo. Polinomi ciclotomici in un campo  $K$ .

Radici primitive in  $K$  e criteri per verificare che lo sono.

Gruppi ciclici.

Teorema di Gauss: Sia  $K$  un campo e sia  $G$  un sottogruppo finito del gruppo moltiplicativo  $K^*$ , allora  $G$  è ciclico. In particolare  $F^*p$  è ciclico per ogni  $p$ .

Sia  $K$  un campo e  $f$  un polinomio irriducibile in  $K[X]$ . Costruzione del campo  $K[X]/\langle f \rangle$ .

Definizione di ideale, ideale principale, ideale massimale. Proprietà'.

Ogni campo finito è isomorfo a un campo del tipo  $F_p[x]/\langle f \rangle$ , con  $p$  caratteristica del campo.

Teorema di esistenza e unicità' dei campi finiti:

dati  $p$  primo e  $n \geq 1$ , esiste un unico campo con  $p^n$  elementi. Più precisamente

Teorema: 1) esiste un polinomio irriducibile in  $F_p[X]$  di grado  $n$

2) se  $F$  e  $F'$  sono campi finiti con  $p^n$  elementi allora  $F$  e  $F'$  sono isomorfi.

Mappa di Frobenius da  $F_p[x]/\langle f \rangle$  in se.

Teorema: Un polinomio  $f$  è irriducibile se e solo se  $\text{Ker}(F)=0$  e  $\text{Ker}(F-I)=F_p$ .

Algoritmo di Berlekamp

Teorema: Il polinomio  $X^{p^n}-X$  in  $F_p$  è il prodotto DEI polinomi monici irriducibili di  $F_p$  di grado  $d$ , con  $0 < d < n$ ;  $d$

$n$  e  $d$  che divide  $n$ . (s.d.)

Corollario: Sia  $N_d$  il numero dei polinomi monici irriducibili di grado  $d$  in  $F_p$ , allora  $p^n = \sum_{d|n} N_d p^{n/d}$ .

Sistemi parzialmente ordinati. Funzione di Moebius. Formule di inversione.

Formula  $N_n = \frac{1}{n} \sum_{d|n} \mu(d) p^{n/d}$  per contare i polinomi monici irriducibili di grado  $n$  in  $F_p$ ,

Definizione di least upper bound e greatest lower bound di un sottoinsieme  $X$  di un insieme  $P$  parzialmente ordinato. Definizione di reticolo. Definizione di reticolo completo. L'esempio dell'insieme delle parti. Il teorema del punto fisso di Tarski: Sia  $L$  un reticolo completo e sia  $f$  una applicazione da  $L$  in  $L$  che conserva l'ordine. Allora l'insieme  $P$  dei punti fissi di  $f$  è un reticolo completo, inoltre il massimo di  $P$  è il least upper bound dell'insieme degli  $x$  in  $L$  tali che  $x \leq f(x)$  e il minimo di  $P$  è il greatest lower bound dell'insieme degli  $x$  in  $L$  tali che  $f(x) \leq x$ . In particolare  $P$  è non vuoto.



**Disciplina:** A000115 **MATEMATICA DISCRETA - Modulo ANALISI** MAT/05

**Corso di Studio:** IIN **Crediti:** 3 **Tipo:** A

**Note:** Modulo del corso Matematica Discreta 6 CFU

**Docente:** VEZZOSI GABRIELE RC MAT/03 **Copertura:** AFF03

**Ente appartenenza:** Dip. Matematica Applicata "G.Sansone"

---

- Generalità sui codici e canali di comunicazione. Distanza di Hamming ed error-detecting error-correcting codes. Esempi.
- Codici lineari. Esempi.
- Il problema fondamentale della teoria dei codici. Alcuni esempi di bounds sui codici. Codici perfetti. Esempi.
- Codici ciclici.

**Disciplina:** 0065208    **PROGETTAZIONE E PRODUZIONE**    ING-INF/05  
**MULTIMEDIALE**

**Corso di Studio:** IIN    **Crediti:** 6    **Tipo:** A

**Note:**

**Docente:** DEL BIMBO ALBERTO    P1    ING-INF/05    **Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

**Disciplina:** N205IIN **SISTEMI DISTRIBUITI**

ING-INF/05

**Corso di Studio:** IIN TES

**Crediti:** 6 **Tipo:** A

**Note:**

**Docente:** NESI PAOLO

P1

ING-INF/05

**Copertura:** AFF03

**Ente appartenenza:** Dip. Sistemi e Informatica

---

Il seguente link portalla pagina del corso per accedere alle slide ed al programma dettagliato:

<http://www.dsi.unifi.it/~nesi/didactical.html>

Influenza la struttura del corso,

vota gli argomenti che ti interessano maggiormente

<http://www.disit.dsi.unifi.it/argomentisdtriennale.html>

--Introduzione: Cosa sono i sistemi distribuiti, Tecnologie dei sistemi distribuiti, Internet e sua Evoluzione, Intranet, Penetrazione di internet, Crescita, Sistemi Mobili, Condivisione delle risorse, Web Server and Web Services, Caratteristiche: Eterogenei, aperti, sicuri, trasparenti, architetture, n-tier.

--Modelli Middleware, pattern architetturali dei sistemi distribuiti, modelli di comunicazione, progettazione, comunicazione fra processi, remote procedure call....

--WEB services e REST (parte 2a, ver:1.2): remote invocation via Web Services and REST architectures.

--CORBA, Architettura CORBA; componenti di CORBA; Oggetti in CORBA; Caratteristiche di CORBA; applicazioni WEB, invocazione via HTML; Costruzione di Applicazioni CORBA; aspetti di Progetto; Single thread e MultiThread

•Sistemi P2P: tecnologiche p2p, requisiti P2P, sicurezza, architetture (distribuite, concentrare e ibride), propagazione/seeding, algoritmi di routing overlay, esempio di Skype, download multisorgente, monitoraggio di sistemi P2P, protocollo bitTorrent, valutazione dei sistemi P2P, esempi: jxta, bittorrent, p2pvod, p2ptv;

--Clock e ordinamenti in sistemi distribuiti P2P: algoritmi per la sincronizzazione di processi, sincronizzazione di orologi/clock, precisione, drift, ritardi, etc.;

--Sistemi GRID: GRID aspetti generali, calcolo parallelo e distribuito, tipi di GRID, problemi dei GRID, protezione dei dati, parallelizzazione degli algoritmi, confronto fra sistemi GRID, media grid, esempi;

--Sistemi Cooperativi, CSCW, Storie di Successo; Tipologie e classificazione, tassonomie, confronti fra sistemi GRID; tecnologie e problemi, progettare un CSCW, Tipi di CSCW e classificazione; tempo e spazio, modelli sincroni ed asincroni, Tecnologie e problemi; Granularità;

--Sistemi Mobili: sistemi mobili, da PDA a cellulari, comunicazione con sistemi mobili; Sistemi operativi per sistemi mobili; applicazioni, profilo di sistemi mobili, problemi di adattamento, sviluppo per sistemi mobili, aspetti di progettazione, esempi di programmazione per windows mobile, e di programmazione per iPhone/iPod;

--Reti Sociali, overview on social network: una breve vista sui problemi delle social network, modelli collaborativi, modellazione delle relazioni, metriche, distanze fra profili, confronto di reti sociali attuali come: youtube, flickr, etc.; Una versione piu' estesa di queste slide e' riportata e fa parte del corso di Laurea Magistrale.

-- Virtualization: creating virtual networks, virtual infrastructure, virtual storage, virtual machines, SAN, Virtual center, security, moving virtual, backup on the virtual, VMware, Vsphere, ESX scaling, MS Hiper-V; fail over, fault tolerance, grid and virtual machines, standard format (OVF, etc.).

