

[Home](#)[Docenti](#)[Corsi](#)[Orario Lezioni](#)[Aule](#)[Appelli](#)[Corsi di insegnamento: Architettura degli Elaboratori](#) Login**Corsi di insegnamento**[Elenco dei corsi](#)[Elenco dei moduli](#)[Cerca](#)**Utilità**[Ricevimento studenti](#)[Registrazione ai corsi](#)[Materiale didattico](#)[Test online](#)[Storico dei corsi](#)[Brochure dei corsi \(pdf\)](#)[Corsi per Facoltà](#)

Architettura degli Elaboratori

Anno accademico 2015/2016**Codice del corso** 13598**Docente** [Prof. Alessandro Dal Palu'](#) (Titolare del corso)**Anno** 1° anno**Corso di studi** L31 Informatica**Tipologia** Caratterizzante**Crediti/Valenza** 6**SSD** INF/01 - informatica**Erogazione** Tradizionale**Lingua** Italiano**Frequenza** Obbligatoria**Valutazione** Scritto ed orale**Periodo didattico** Primo semestre**Storico** [Anni precedenti](#) ▼

Obiettivi formativi del corso

Proporre un percorso che, partendo dalla rappresentazione dell'informazione e passando per lo studio delle reti logiche, arrivi a definire i fondamentali aspetti architetturali dei calcolatori elettronici.

Il corso prevede, oltre a lezioni teoriche, una serie di esercitazioni in aula sulla parte di reti logiche e in laboratorio sulla parte di assembly IA-32.

Con riferimento agli Indicatori di Dublino:

Conoscenza e capacità di comprensione

Il corso introduce i primi concetti relativi all'architettura degli elaboratori. Particolare enfasi è data alla comprensione delle architetture classiche basate sul Modello di Von Neumann. Il testo di riferimento è in italiano, ma viene altresì utilizzata durante le lezioni la terminologia in lingua inglese come avviamento alla consultazione di letteratura scientifica internazionale.

Capacità di applicare conoscenza e comprensione

Le conoscenze teoriche presentate vengono sempre applicate alla risoluzione di problemi specifici. Le esercitazioni che affiancano il corso sono incentrate sulla risoluzione di esercizi e problemi, con particolare riferimento all'interfaccia tra hardware e software. Spesso i metodi risolutivi vengono presentati sotto forma algoritmica, sviluppando negli studenti la capacità di strutturare procedure utili in numerose parti dell'informatica e non solo nello studio dell'architettura degli elaboratori.

Autonomia di giudizio

Gli esercizi, che vengono proposti relativamente alla parte teorica svolta a lezione, possono venire risolti individualmente o in gruppo. Il confronto con i compagni di corso, nel lavoro a casa o durante gli svolgimenti in aula, favorisce lo sviluppo di capacità specifiche per poter a chiarire ai compagni o ai docenti le proprie argomentazioni. Spesso gli esercizi proposti possono venire risolti in modi molto diversi e l'ascolto delle soluzioni proposte da altri permette di

sviluppare la capacità di individuare strutture comuni, al di là delle apparenti differenze superficiali.

Abilità comunicative

Le numerose discussioni sui diversi metodi per risolvere i problemi proposti consentono di migliorare le capacità di comunicazione. Vengono inoltre abitualmente utilizzate durante le spiegazioni (ed esplicitamente evidenziate in classe) alcune modalità di comunicazione specifiche della tecnologia informatica.

Capacità di apprendimento

Lo studio delle origini delle soluzioni tecnologiche e la loro introduzione motivata da considerazioni quantitative contribuisce a realizzare negli studenti la capacità di apprendere in modo profondo e non soltanto superficiale e ripetitivo. Le conoscenze così acquisite non sono mai rigide e definitive, ma sono perfettamente adattabili ad ogni evoluzione e cambiamento di prospettiva e di contesto.

Risultati dell'apprendimento

Risultati dell'apprendimento e modalità di verifica

Saper comprendere e utilizzare in modo appropriato le tecniche relative alla rappresentazione dell'informazione. Saper lavorare a livello logico nella progettazione di semplici reti combinatorie e sequenziali sincrone. Conoscere l'architettura di base degli elaboratori secondo il Modello di Von Neumann sia a livello funzionale che di micro-architettura. Conoscere e saper utilizzare le principali tecniche di programmazione in linguaggio assembly per architetture classiche.

Modalità di verifica/esame

L'esame consiste in una prova scritta e un colloquio orale a cui si accede solo al superamento della prova scritta. E' possibile sostenere più volte la prova scritta ma ogni scritto consegnato annulla lo scritto precedente.

Note

Per il materiale didattico e il programma dettagliato seguito a lezione cliccare qui:

<http://www2.unipr.it/~dalpalu/corsi/arch/arch1415.html>

Programma

I parte – Introduzione ai sistemi di elaborazione

- Sistemi di elaborazione
- Evoluzione storica e tecnologica
- Architettura e organizzazione dei calcolatori elettronici
- Modelli di sistema e livelli di astrazione
- Sistemi di numerazione binario, esadecimale e ottale
- Rappresentazione binaria dei numeri reali
- Informazioni di carattere alfanumerico
- Rappresentazione di suoni, immagini e video

II parte – Livello logico

- Algebra delle reti
- Circuiti logici elementari
- Forme canoniche e trasformazioni
- Minimizzazione logica
- Reti combinatorie
- Reti sequenziali

III parte – Livello funzionale

- Relazione tra livello funzionale e livello micro-architettura
- Organizzazione di un calcolatore moderno
- Il repertorio delle istruzioni
- La CPU
- I sistemi di memoria
- Il sottosistema di I/O

IV parte – Architettura e linguaggio assembly IA-32

- Architettura IA-32
- Le CPU IA-32: gestione della memoria, registri, flag
- Istruzioni dell'assembly IA-32
- Set di istruzioni aggiuntive dei moderni microprocessori: MMX, 3DNow!, SSE, SSE2
- Programmazione assembly ed interfaccia con il linguaggio C

V parte – Livello software

- Cenni al sistema operativo
- Cenni alla struttura dei compilatori di linguaggi tipo-C
- Cenni alla generazione di codice assembly per un sotto-insieme del C

VI parte – Livello micro-architettura

- La microarchitettura: progettazione hardwired e micro-programmata
- Approfondimenti sulle architetture micro-programmate
- Cenni alla struttura di una architettura micro-programmata compatibile con le CPU tipo NMOS 6502
- Gestione del I/O a livello micro-architettura: interrupt e DMA
- Interazione con il livello software

Testi consigliati e bibliografia

1. Giacomo Bucci. *Architettura e organizzazione dei calcolatori elettronici - Fondamenti*, McGraw-Hill, 2004
2. David A. Patterson, John L. Hennessy. *Struttura e Progetto dei Calcolatori*, Zanichelli, 2006.
3. William Stallings. *Architettura e organizzazione dei calcolatori - Progetto e prestazioni*, Addison Wesley, 2004.
4. Andrew S. Tanenbaum. *Architettura dei Calcolatori*, 5a edizione, Prentice Hall, 2006.
5. Randall Hyde. *The Art of Assembly Programming*, disponibile online.

Orario lezioni ▼

Giorni	Ore	Aula
--------	-----	------

Registrazione  Attiva



[Materiale didattico](#)



Test online



Vai a Moodle



Visita i forum

Ultimo aggiornamento: 20/10/2015 16:35

