

Testi del Syllabus

Resp. Did.	PRATI Andrea	Matricola: 017401
Anno offerta:	2016/2017	
Insegnamento:	1006620 - ARCHITETTURA DEI CALCOLATORI ELETTRONICI	
Corso di studio:	3050 - INGEGNERIA INFORMATICA, ELETTRONICA E DELLE TELECOMUNICAZIONI	
Anno regolamento:	2014	
CFU:	9	
Settore:	ING-INF/05	
Tipo Attività:	B - Caratterizzante	
Anno corso:	3	
Periodo:	Secondo Semestre	
Sede:	PARMA	



Testi in italiano

Lingua insegnamento

Italiano

Contenuti

Il corso fornisce i concetti di base delle architetture dei sistemi di elaborazione, analizzandone l'organizzazione interna e descrivendo i principi di funzionamento delle CPU, delle memorie, delle strutture di interconnessione e di ingresso/uscita. Il corso comprende le nozioni fondamentali della programmazione in linguaggio assembly. I concetti sono esposti in modo generale; quando necessario si fa diretto riferimento all'architettura dei processori INTEL.

Programma

Introduzione e storia dei calcolatori elettronici:

- * Macchina di Turing
- * Architettura di Von Neumann

Introduzione alle architetture delle moderne CPU:

- * RISC vs CISC
- * Cenni ai processori RISC con esempi
- * Microarchitettura della CPU
- * Architettura monociclo
- * Architettura multiciclo

Architetture avanzate:

- * Pipeline e alee
- * Architetture superscalari e gestione conflitti

Memorie:

- * Generalità, memorie a semiconduttore (ROM e RAM)
- * Memorie secondarie: tecnologie e prestazioni.
- * Gerarchie di memoria
- * Principio di località e concetti generali
- * Memoria virtuale (segmenti e pagine)
- * Memoria cache (tecniche di accesso, politiche di sostituzione, ...)

Sottosistema di ingresso uscita:

- * Gestione a controllo di programma
- * Gestione sotto controllo di interrupt
- * DMA

Introduzione all'assembly dell'architettura x86

- * Architettura logica delle CPU x86.

- * Sintassi. Modi di indirizzamento. Operazioni e pseudo-operazioni.
- * Funzioni DOS e BIOS .
- * Esempi di programmi in assembly.

Testi di riferimento

- A.S. Tanenbaum e T. Austin, "Architettura dei Calcolatori: un approccio strutturale", sesta ed., Pearson/Prentice Hall, 2013
- Giacomo Bucci, "Calcolatori elettronici – Architettura e organizzazione", Mc Graw-Hill, 2009
- William Stallings, "Architettura e organizzazione dei calcolatori. Progetto e prestazioni", ottava ed., Pearson Prentice-Hall, 2010
- Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Introduzione all'architettura dei calcolatori", seconda ed., McGraw-Hill, 2007

Obiettivi formativi

Conoscenza e Comprensione

Lo scopo del corso quello di illustrare i meccanismi di base di funzionamento dei sistemi di calcolo e in particolare l'interazione fra i moduli (CPU, memoria, I/O) di un calcolatore.

Capacità di applicare conoscenza e comprensione

Lo studente acquisirà la capacità di comprendere il funzionamento di base delle moderne CPU.

Lo studente al termine del corso sarà capace a scrivere anche semplici programmi a livello assembler.

Metodi didattici

Il corso comprende indicativamente 40 ore di lezioni tradizionali in aula e 24 ore di esercitazione in laboratorio.

Modalità di verifica dell'apprendimento

L'esame consiste in due parti, che possono essere date indipendentemente, sia come sessione sia come ordine. La prima parte è una prova orale fatta per iscritto, su domande di teoria e un esercizio sulle memorie cache. La seconda parte è una prova pratica in laboratorio che richiede lo sviluppo di un programma in Assembly x86. Il voto finale è dato dalla media dei due voti.



Testi in inglese

Lingua insegnamento

Italian

Contenuti

The purpose of the course is to give the students the basic concepts of the modern CPU architectures and computer systems. The course comprises the fundamental knowledge of the programming in language Assembly. The concepts are exposed in a general way; when necessary it is made directed reference the architecture of the INTEL CPU family.

Program

Introduction and history of computers:

- * Turing machine
- * Von Neumann architecture

Introduction to the architecture of the modern CPUs.

- * RISC vs CISC
- * Brief introduction to RISC processor with examples
- * CPU micro-architecture
- * Monocycle architecture
- * Multicycle architecture

Advanced architectures:

- * Pipeline and hazards
- * Superscalar architectures and conflict management

Memories:

- * Basic principles, semiconductor memories (ROM and RAM)
- * Secondary memories: technologies and performance
- * Memory systems hierarchy
- * Locality concept
- * Cache memories (address mapping, line size, replacement and writeback policies)

* Virtual memory systems.
Input/output subsystem:
* Polling
* Interrupt
* DMA
Introduction to the Assembly language for x86 architecture:
* Logical architecture of x86 CPUs
* Syntax. Addressing modes. Operations and pseudo-operations.
* DOS and BIOS functions
* Examples of programs in Assembly

Testi di riferimento

- A.S. Tanenbaum e T. Austin, "Architettura dei Calcolatori: un approccio strutturale", sesta ed., Pearson/Prentice Hall, 2013
- Giacomo Bucci, "Calcolatori elettronici – Architettura e organizzazione", Mc Graw-Hill, 2009
- William Stallings, "Architettura e organizzazione dei calcolatori. Progetto e prestazioni", ottava ed., Pearson Prentice-Hall, 2010
- Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Introduzione all'architettura dei calcolatori", seconda ed., McGraw-Hill, 2007

Obiettivi formativi

Knowledge and understanding
The purpose of the course to illustrate the basic mechanisms of operation of computing systems and in particular the interaction between the modules (CPU, memory, I / O) of a computer.

Applying knowledge and understanding
The student will acquire the ability to understand the basic functioning of modern CPUs.
Students completing the course will be able to write simple programs in Assembly.

Metodi didattici

The course includes around 40 hours of traditional frontal lectures and 24 hours of training in the laboratory.

Modalità di verifica dell'apprendimento

The exam consists in two parts, which can be taken independently, in terms of both order and exam session. The first part consists in an oral examination made in written, with questions about the theory and an exercise on cache memories. The second part is a practical exam in laboratory requiring to write a program in Assembly x86. The final grade will be the average of the two grades.