

Guida rapida all'insegnamento

Architettura degli elaboratori

Docente: Giuseppe Scollo

Università di Catania
Facoltà di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea in Informatica, A-A 2010-11

1 di 12

Indice

1. Guida rapida all'insegnamento
2. Organizzazione dell'insegnamento
3. Obiettivi formativi
4. Attività formative
5. Modalità di valutazione
6. Programma delle lezioni (1)
7. Programma delle lezioni (2)
8. Testi consigliati
9. Testi per consultazione
10. Simulatori
11. Interazioni

2 di 12

Organizzazione dell'insegnamento

- modalità: *blended e-learning*
- problemi ed esercizi sugli argomenti delle lezioni, ed eventuali altri materiali didattici, via via disponibili in rete
- proposta: usare la piattaforma di e-learning dell'ateneo per problemi ed esercizi
- sviluppo progettuale cooperativo
del progetto formativo dell'insegnamento
- condivisione di obiettivi formativi

3 di 12

Obiettivi formativi

Acquisizione e sviluppo della capacità di:

- comprendere i concetti fondamentali di architettura dei sistemi di calcolo e i principi metodologici che ne ispirano lo sviluppo, nella prospettiva storica della loro evoluzione
- adoperare, progettare e realizzare strumenti software, quali simulatori ed interpreti, per macchine astratte relative ai livelli più bassi della organizzazione dei sistemi di calcolo

4 di 12

Attività formative

L'organizzazione dell'insegnamento prevede 72 ore di lezione (ed esercitazione)

L'acquisizione di concetti e metodi nella disciplina è sostenuta da:

- frequenza delle lezioni ed esercitazioni
- studio del testo di riferimento e delle note integrative
- consultazione di altri testi e materiali didattici
- sperimentazione pratica con simulatori e altri strumenti software
- elaborazione di soluzioni a problemi ed esercizi proposti:
v. area Test sottostante
- interazione con il docente (ricevimento settimanale)
- interazione collaborativa con i colleghi (e il docente)

5 di 12

Modalità di valutazione

- obiettivi della valutazione
- prova scritta e verifica orale:
valutazione del conseguimento degli obiettivi formativi essenziali
- colloquio opzionale:
valutazione della maturità concettuale e scientifica nella disciplina
- prova scritta
- alcuni quesiti a risposta chiusa (~50-70%)
- qualche quesito a risposta aperta, con limiti di spazio (~30-50%)
- esame orale
- verifica individuale delle motivazioni di risposte a quesiti della prova scritta
- eventuale colloquio (opzionale) individuale sugli argomenti del programma, a partire da un elaborato originale su un progetto software concordato con il docente, che può essere realizzato in collaborazione con altri studenti
- prova in itinere
- anticipazione della prova scritta, autovalutazione finale (valida per l'esame)

Il superamento dell'esame porta all'acquisizione di 9 crediti.

N.B. le stesse modalità di valutazione si applicano all'esame da 6 CFU, per studenti dell'AA 2007-08, ma su un programma ridotto, che esclude dalla valutazione gli argomenti delle seguenti lezioni: 6 (escluse solo le note integrative 2), 14, da 19 a 22

6 di 12

Programma delle lezioni (1)

1. Cenni storici sulle macchine da calcolo, macchine astratte
Tanenbaum, cap. 1 fino a 1.2.6 + Note int. 1
2. Tassonomia dei sistemi di calcolo
Tanenbaum, cap. 1 da 1.3 alla fine
3. Principi di progettazione di processori e memorie
Tanenbaum, cap. 2 fino a 2.2.6
4. Dispositivi di memoria secondaria
Tanenbaum, cap. 2 da 2.3 a 2.3.11
5. Dispositivi di ingresso e uscita
Tanenbaum, cap. 2 da 2.4 alla fine
6. Strutture algebriche, algebre di Boole
Note int. 2 + Tanenbaum, cap. 3 fino a 3.1.4
7. Livello logico-digitale, porte e circuiti logici, reti combinatorie
Note int. 3 + Tanenbaum, cap. 3 da 3.2 a 3.2.4
8. Circuiti sequenziali, chip di memoria e della CPU
Tanenbaum, cap. 3 da 3.3 a 3.4.1
9. Bus di sistemi di calcolo
Tanenbaum, cap. 3 da 3.4.2 a 3.6.4, escl. 3.5.2, 3.5.3, 3.6.2 da "Segnali del bus PCI" in poi
10. Concetti di microarchitettura
Tanenbaum, cap. 4 fino a 4.1.3
11. Linguaggi di microprogrammazione e ISA
Note int. 4 + Tanenbaum, cap. 4 da 4.2 a 4.3.1
12. Microprogrammazione di interpreti ISA
Tanenbaum, cap. 4, 4.3.2

7 di 12

Programma delle lezioni (2)

13. Progettazione della microarchitettura
Tanenbaum, cap. 4 da 4.4 a 4.4.3
14. Miglioramento delle prestazioni di microarchitetture
Tanenbaum, cap. 4 da 4.4.4 a 4.5.4
15. Architettura ISA, progettazione di formati d'istruzioni
Tanenbaum, cap. 5 fino a 5.1.5 e 5.3 fino a 5.3.3 escl. 5.3.2
16. Modalità d'indirizzamento, trap, I/O e gestione delle interruzioni
Tanenbaum, cap. 5: 5.4 fino a 5.4.11 escl. 5.4.8 e 5.4.9, cap. 3: 3.7, cap. 5: 5.5.7, 5.6.4, 5.6.5 + Note int. 5, 6
17. Linguaggio assemblativo, macroistruzioni
Tanenbaum, cap. 7 fino a 7.2.4
18. Processo di assemblaggio, collegamento e caricamento
Tanenbaum, cap. 7 da 7.3 alla fine
19. Memoria virtuale
Tanenbaum, cap. 6 fino a 6.1.10 escl. 6.1.8 e 6.1.9
20. Istruzioni virtuali di I/O e per il calcolo parallelo
Tanenbaum, cap. 6 da 6.2 a 6.3.3
21. Chip di calcolo parallelo e coprocessori
Tanenbaum, cap. 8 fino a 8.2.1
22. Multiprocessori, reti di calcolo, griglie computazionali
Tanenbaum, cap. 8 da 8.3 a 8.3.2, 8.4 fino a 8.4.1, 8.4.4, 8.4.5, 8.4.7, 8.5
23. Riepilogo

8 di 12

Testi consigliati

Testo di riferimento

Andrew S. Tanenbaum : *Architettura dei calcolatori: un approccio strutturale*
Quinta edizione, Edizione italiana, Pearson Prentice Hall (2006)

con le correzioni identificate finora nell'ambito dell'insegnamento

Note integrative

1. Cenni storici sulle macchine da calcolo (G. Scollo)
 2. Strutture algebriche, algebre di Boole (G. Scollo)
 3. Note introduttive sul livello della logica digitale (F. Barbanera, 2008)
 4. Note introduttive sul linguaggio Mic-1 (F. Barbanera, 2008)
 5. Un esempio di Trap per IJVM (F. Barbanera, 2008)
 6. Esempio (con simulatore) di realizzazione di I/O con interruzioni
- altre note disponibili nel prosieguo delle lezioni

Testi per consultazione

Libri di testo

le parti non trattate del testo di Tanenbaum sono consultabili per approfondimenti

G. Bucci : *Calcolatori elettronici - Architettura e organizzazione*, McGraw-Hill (2009)

C. Hamacher, Z. Vranesic, S. Zaky : *Introduzione all'architettura dei calcolatori*, 2/e, McGraw-Hill (2007)

J.L. Hennessy, D.A. Patterson : *Architettura degli elaboratori*, Apogeo (2008)

D.A. Patterson, J.L. Hennessy : *Struttura e progetto dei calcolatori*, 3/e, Zanichelli (2010)

W. Stallings : *Architettura e organizzazione dei calcolatori*, 8/e, Pearson (2010)

Note integrative per consultazione

Macchine astratte e struttura a livelli dei computer moderni

in: N. Fazio, A. Nicolosi e F. Barbanera, *Introduzione alle Macchine Astratte*

Attualità e pratica dell'aritmetica Maya, A. D'Agata, B. Radelli, G. Scollo, IDD'97.

Note sul problema del bit alto dell'istruzione Mic-1 (corrispondenza personale, 2011).

Lezioni di Wolfgang Schreiner (2005, pagina in tedesco, lucidi in inglese).

Manuali delle architetture Intel® 64 e IA-32.

Manuali delle architetture SPARC® (pagina di download).

altri materiali saranno indicati nel prosieguo delle lezioni (nel frattempo molti materiali sono reperibili presso il sito della precedente edizione dell'insegnamento)

Simulatori

i simulatori con numero di versione (più recente) sono stati realizzati nell'edizione corrente dell'insegnamento, gli altri in edizioni precedenti, ma utilizzati in lezioni dell'edizione corrente

- Conversione della rappresentazione dei numeri su abaco Maya (v. 1.0, Java: R. Carrubba)
 - Operazioni aritmetiche additive su abaco Maya (v. 1.0, Java: S. Alparone, P. Catania)
 - Simulatore di codice di Hamming (v. 1.0, Java: F. Bombace)
 - Ordinamento dei byte (v. 1.0, VisualBasic, MS Windows: V. Mansueto)
 - Simulatore di porte logiche (v. 1.0, Java: D. Catalano, G. Gelardi, M. Quattrocchi, F. Trenta)
 - Arbitraggio del bus (Java: R. Capuano)
 - Simulatore di ALU a 8 bit (v. 1.0, Flash: S. Lentini e G. Nicotra)
 - Simulatore di ALU a 8 bit (v. 1.0, Java: F. Vindigni)
 - Simulatore di PLA (v. 1.0, Python: A. Plebe)
 - Emulatore Mic-1, con traduttore MAL→Mic-1 (v. 1.3, Java: M.R. Aiello, v. 1.2: D. Spitaleri, E. Torre, F. Ravi)
 - Convertitore istruzioni Mic-1→MAL (HTML con Javascript: D. Vacalluzzo)
 - Convertitore Mic-1→MAL (v. 2.0, Java, Android: E. Martorana)
 - Struttura della macchina Mic-1 (Flash: S. Nativo)
 - Simulatore di Stack in InvokeVirtual/IReturn (Flash: F. Tendi)
 - Simulatori di Stack JVM nell'esecuzione di metodi ricorsivi (Java: D. Porrovecchio)
 - Simulatore di esecuzione di programma assembly Torri di Hanoi (v. 1.0, Java: S. Cali)
 - Emulatore (con assembler) della macchina JVM (Java: D. Spitaleri)
 - Esempio, con simulatore, di realizzazione di I/O con interruzioni (HTML con Java applet)
 - Simulatore di allocazione di blocchi a file su disco (v. 1.0, Java: J. Catania)
 - Simulatore di memoria virtuale (v. 1.0, Java: D. Lanza)
- altri simulatori disponibili dalle precedenti edizioni dell'insegnamento.

11 di 12

Interazioni

Forum, Dokeos Test: cosa va dove?

- Forum: discussioni di
 - tematiche pertinenti alle lezioni
 - soluzioni di problemi a risposta libera
 - organizzazione dell'insegnamento
 - problemi con l'uso di simulatori, etc.
- Dokeos Test:
 - sviluppo di problemi ed esercizi proposti
 - discussione ed elaborazione di idee di progetti software (discussioni ed aree di lavoro a visibilità limitata)

12 di 12