UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Computer Architecture

2223-1-E3101Q104

Obiettivi

Alla fine del corso lo studente avrà conoscenza degli elementi dell’architettura di un semplice elaboratore e delle basi della programmazione assembly, abilità di progettare piccole modifiche alla struttura interna di un calcolatore e di scrivere semplici programmi assembly, e infine competenza nel valutare le tecnologie più adeguate, in termini di prestazioni, per diversi ambiti di elaborazione.

Contenuti sintetici

- Principali elementi dell’architettura hardware di un elaboratore.
- Instruction set architecture.
- Catena programmativa.
- Controllo del percorso dei dati.
- Gestione delle eccezioni.
- Tecniche di gestione dell'ingresso/uscita.
- Gerarchie di memoria: cache.

Programma esteso

1. Rappresentazione del'informazione

- rappresentazione dell'informazione non numerica,
- rappresentazione dei numeri interi con e senza segno,
- rappresentazione dei numeri in virgola fissa e mobile,
- +: dettagli dei calcoli in virgola mobile con IEEE754.

3. Circuiti logici

- reti combinatorie,
- reti sequenziali e FSM (Finite State Machine),
- rassegna di circuiti notevoli (decoder, multiplexer, register file, ALU, etc.).

5. Instruction Set Architecture

- schema di von Neumann,
- CPU, registri, ALU e memoria,
- ciclo fondamentale di esecuzione di una istruzione (fetch/decode/execute),
- tipi e formati di istruzioni MIPS32,
- modalità di indirizzamento.

7. Linguaggio Assembly

- formato simbolico delle istruzioni,
- catena di programmazione (compilatore, assembler, linker, loader, etc.),
- pseudo-istruzioni e direttive dell'assemblatore,
- scrittura di semplici programmi assembly,
- convenzioni programmative (memoria, nomi dei registri, etc.).

9. Datapath

- percorsi dei dati per le diverse classi di istruzioni,
- controllo del percorso dei dati con FSM (implementazione multi-ciclo).
- +: pipelining e gestione hazard

11. Gestione delle eccezioni

- tassonomia di eccezioni in terminologia MIPS32,
- modifiche alla FSM di controllo, registro Cause, etc.

13. Tecniche di gestione dell'ingresso/uscita

- controllo di programma,
- interruzione di programma,
- accesso diretto alla memoria.

15. Gerarchie di memoria: cache

- cache a mappatura diretta,
- cache fully associative,
- cache n-way set associative
- +: rimpiazzamento con LRU.

Prerequisiti

Nessuno

Modalità didattica

- lezioni
- esercitazioni
- laboratorio
- studio individuale

Le lezioni sono tenute in italiano e come usuale in aula.

Materiale didattico

- Libro di testo: David Patterson, John Hennessy: Computer Organization and Design, The Hardware/Software Interface, 5th edition, Morgan Kaufmann (Elsevier) / "Struttura e progetto dei calcolatori" Zanichelli
- Materiale disponibile su elearning relativo a lezioni, esercitazioni e laboratorio, alcune prove di autovalutazione, etc.

Periodo di erogazione dell'insegnamento

Secondo semestre

Modalità di verifica del profitto e valutazione

L'esame potrà essere svolto in due prove parziali, tenute una a metà e una a fine semestre, o come sempre in un unico appello tra quelli a calendario lungo tutto l'anno accademico.

Ogni prova prevede una sessione al calcolatore, comprendente una parte obbligatoria ed una facoltativa. Il voto pieno si ottiene solo svolgendo anche la parte facoltativa.

La prima parte è costituita da domande a risposte chiuse (prevalentemente esercizi) e aperte, e verte su un insieme ridotto di argomenti, definiti come "Non Eludibili".

La seconda parte è a risposte aperte e verrà valutata solo per chi ha riportato un esito sufficiente nella prima parte. Questa parte è di approfondimento e verte anche su argomenti non inclusi nella prima parte, indicati con '+' nel programma in questo syllabus.

Orario di ricevimento

Inviare email per concordare un appuntamento

Sustainable Development Goals