



UNIVERSITÀ DEGLI STUDI DI PALERMO

DIPARTIMENTO	Matematica e Informatica		
ANNO ACCADEMICO OFFERTA	2023/2024		
ANNO ACCADEMICO EROGAZIONE	2023/2024		
CORSO DILAUREA	INFORMATICA		
INSEGNAMENTO	ARCHITETTURE DEGLI ELABORATORI		
TIPO DI ATTIVITA'	A		
AMBITO	50168-Formazione informatica di base		
CODICE INSEGNAMENTO	16450		
SETTORI SCIENTIFICO-DISCIPLINARI	INF/01		
DOCENTE RESPONSABILE	BELLAVIA FABIO	Ricercatore a tempo determinato	Univ. di PALERMO
ALTRI DOCENTI			
CFU	6		
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	86		
NUMERO DI ORE RISERVATE ALLA DIDATTICA ASSISTITA	64		
PROPEDEUTICITA'			
MUTUAZIONI			
ANNO DI CORSO	1		
PERIODO DELLE LEZIONI	2° semestre		
MODALITA' DI FREQUENZA	Facoltativa		
TIPO DI VALUTAZIONE	Voto in trentesimi		
ORARIO DI RICEVIMENTO DEGLI STUDENTI	BELLAVIA FABIO Giovedì 14:00 16:00 DMI - Stanza 203		

PREREQUISITI	Conoscenze elementari di programmazione e di matematica discreta.
RISULTATI DI APPRENDIMENTO ATTESI	<p>Conoscenza e capacità di comprensione: acquisizione e comprensione dei fondamenti scientifici e tecnologici relativi alle architetture dei sistemi di elaborazione, dei diversi livelli e moduli che li compongono, delle loro relazioni a diversi livelli di astrazione. Acquisizione e comprensione delle basi dei linguaggi di descrizione hardware e degli strumenti di sviluppo per la progettazione, l'analisi, la simulazione e la sintesi di circuiti digitali. Acquisizione e comprensione delle basi della programmazione a basso livello tramite assembly. Conoscenza e comprensione delle analogie e differenze tra programmazione di basso livello tramite assembly e programmazione di alto livello in C, e delle loro possibili interazioni e integrazioni.</p> <p>Capacità di applicare conoscenza e comprensione: capacità di utilizzare le conoscenze acquisite per la progettazione, l'analisi, la simulazione e l'ottimizzazione di moduli e unità di controllo a diversi livelli dell'architettura degli elaboratori, anche per mezzo dei linguaggi di descrizione hardware. Capacità di valutare le prestazioni dei sistemi di elaborazione. Capacità di utilizzare il linguaggio assembly per la programmazione a basso livello o per l'ottimizzazione di porzioni di codice.</p> <p>Autonomia di giudizio: capacità di saper valutare e selezionare in base alle esigenze di sviluppo le tecniche di progettazione di circuiti, i componenti architettonici e l'implementazione software più adatti allo scopo.</p> <p>Abilità comunicative: conoscenza dei formalismi e della terminologia per la descrizione di circuiti, dei componenti dei calcolatori, e della programmazione a basso livello.</p> <p>Capacità di apprendimento: capacità di apprendere, confrontare e approfondire tramite le conoscenze acquisite gli sviluppi più recenti o ulteriori modelli architettonici alternativi.</p> <p>Competenze acquisite: utilizzo di strumenti di CAD e dei linguaggi di descrizione hardware per la progettazione e la simulazione di semplici circuiti e blocchi costruttivi digitali. Capacità di analisi e valutazione del comportamento di un segnale dal suo diagramma temporale. Capacità di scrivere programmi in assembly, codificare istruzioni in linguaggio macchina e analizzare tramite gli strumenti specifici i file eseguibili binari o il codice oggetto relativo. Capacità di interfacciare e trasporre codice di alto livello in C con codice assembly e viceversa. Capacità di saper calcolare il tempo medio di accesso alla memoria per differenti configurazioni della gerarchia della memoria.</p>
VALUTAZIONE DELL'APPRENDIMENTO	<p>L'esame consisterà di una prova scritta seguita da una orale. La prova scritta verterà sugli argomenti trattati nel corso, e conterrà quesiti di tipo teorico e brevi esercizi di tipo applicativo, posti in forma di domande a risposta aperta o multipla. La durata prevista della prova è di tre ore. Gli studenti che risulteranno dalla prova scritta aver raggiunto un livello di conoscenza sufficiente della materia saranno ammessi a sostenere l'esame finale orale. La prova orale sarà volta a valutare eventuali lacune da parte dello studente emerse dalla prova scritta, così come la conoscenza di argomenti non presenti in quest'ultima.</p> <p>Il giudizio finale verrà formulato in accordo ai seguenti criteri, tenendo conto del livello delle conoscenze acquisite e delle capacità di applicare tali conoscenze:</p> <ul style="list-style-type: none"> - Eccellente (28-30 / 30L): ottima conoscenza e comprensione degli argomenti trattati, padronanza della terminologia appropriata, capacità di applicare efficacemente in maniera autonoma le conoscenze acquisite; - Buono (23-27): buona conoscenza e comprensione degli argomenti trattati, uso corretto della terminologia, capacità di applicare le conoscenze acquisite in maniera corretta; - Sufficiente (18-22): conoscenza e comprensione basilare degli argomenti trattati, minime conoscenze della terminologia appropriata, scarse capacità di applicare le conoscenze acquisite; - Insufficiente: il livello di conoscenza raggiunto della materia non è accettabile.
OBIETTIVI FORMATIVI	<p>Il corso mira a fornire la conoscenza dei fondamenti scientifici e tecnologici delle architetture dei sistemi di elaborazione, e le conoscenze e le abilità di base necessarie al progetto e alla programmazione degli elementi costruttivi degli elaboratori:</p> <ul style="list-style-type: none"> - I fondamenti della progettazione dei circuiti digitali; - La conoscenza dell'organizzazione dell'architettura dei sistemi di elaborazione; - La capacità di sviluppare componenti a diversi livelli dell'architettura dei sistemi di elaborazione; - La conoscenza dei linguaggi di descrizione hardware e la capacità di saper utilizzare gli strumenti di simulazione; - La conoscenza delle tecniche e degli strumenti di analisi e sviluppo per la programmazione a basso livello;

	- La capacità di interfacciare e trasporre codice di alto livello in C con codice assembly e viceversa.
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali in aula e in laboratorio.
TESTI CONSIGLIATI	<p>- Testo principale / Main suggested book David Money Harris, Sarah L. Harris, "Sistemi digitali e architettura dei calcolatori - Progettare con tecnologia ARM", Zanichelli, 2017, ISBN: 9788808920737</p> <p>- Testi di consultazione / Reference books: M. Morris Mano, Charles Kime, Tom Martin, "Reti logiche", 5/Ed., Pearson, 2019, ISBN: 9788891905819 John L. Hennessy, David A. Patterson "Struttura e progetto dei calcolatori 5/Ed.", Zanichelli, 2022, ISBN: 9788808399854 Robert G. Plantz, "Introduction to Computer Organization: ARM Assembly Language Using the Raspberry Pi", 2021, ebook online gratuito (https://bob.cs.sonoma.edu/IntroCompOrg-RPi/intro-co-rpi.html) John L. Hennessy, David A. Patterson, "Computer Architecture: A Quantitative Approach", 6/Ed, Morgan Kaufmann, 2017, ISBN: 9780128119051 Francesco Berto, "Logica da zero a Gödel", 7/Ed, Laterza, 2008, ISBN: 9788842086345</p>

PROGRAMMA

ORE	Lezioni
1	Livelli di astrazione e gestione della complessità nei sistemi digitali e nell'architettura dei calcolatori.
2	Rappresentazione e codifica binaria dei numeri interi.
2	Porte logiche, livelli logici e margini di rumore, transistor CMOS, consumo di potenza.
2	Reti logiche combinatorie, algebra booleana, semplificazione di equazioni booleane, forme canoniche, convenzioni per la rappresentazione di circuiti digitali, blocchi costruttivi combinatori: multiplexer e decoder, temporizzazione di circuiti combinatorici.
3	Reti logiche sequenziali, latch e flip-flop, macchine a stati finiti, temporizzazione di circuiti sequenziali, metastabilità e sincronizzazione, parallelismo spaziale e temporale.
4	Ulteriori blocchi costruttivi digitali: circuiti aritmetici, ALU e flag di stato, contatori e registri a scorrimento.
2	Rappresentazione dei numeri in virgola fissa e mobile, il formato IEEE 754.
3	Componenti di memoria: memorie statiche e dinamiche ad accesso casuale, memorie di sola lettura, FG MOS, memorie flash, reti logiche tramite componenti di memoria; matrici logiche: PLA e FPGA.
7	La microarchitettura ARM: progettazione e analisi delle prestazioni di processori a singolo ciclo, multiciclo e pipeline; componenti hardware, percorso dati e unità di controllo; fasi dell'elaborazione delle istruzioni; gestione delle dipendenze nei processori multiciclo.
3	Microarchitetture avanzate: pipeline lunghe, micro-operazioni, previsione dei salti, processori superscalari e out-of-order, ridenominazione dei registri, multithreading e multiprocessori.
3	Sistemi di memoria: implementazione e gestione della cache e della memoria virtuale.

ORE	Laboratori
2	Richiami alla programmazione in C, implementazione di funzioni in C per la manipolazione di interi in formato binario.
3	Progettazione e implementazione di macchine a stati finiti di Moore e Mealy.
5	Utilizzo di PSpice per la progettazione e simulazione di semplici reti logiche combinatorie e sequenziali, e dei principali blocchi costruttivi digitali.
8	Linguaggi di descrizione hardware: simulazione e sintesi, modellazione comportamentale e strutturale; utilizzo del VHDL per descrivere e simulare tramite il simulatore GHDL reti logiche combinatorie e sequenziali, i principali blocchi costruttivi digitali e i componenti di memoria.
3	L'architettura ARM: il set di istruzioni ARM e la sua codifica; struttura di un programma in memoria: le fasi della generazione di un codice eseguibile, il formato dei file eseguibili e dei file oggetto.
8	Sviluppo di programmi in ARM assembly tramite lo GNU ASsembler (GAS): trasposizione in ARM assembly dei principali costrutti di programmazione e delle principali strutture dati in C, convenzioni per la chiamata a funzione.
3	Approfondimenti sull'architettura ARM: integrazione e interfacciamento tra codice C e ARM assembly, semplici ottimizzazioni del codice in ARM assembly, trasposizione di codice C in ARM assembly.