



UNIVERSITÀ DEGLI STUDI DI PALERMO

DIPARTIMENTO	Ingegneria
ANNO ACCADEMICO OFFERTA	2018/2019
ANNO ACCADEMICO EROGAZIONE	2018/2019
CORSO DILAUREA	INGEGNERIA INFORMATICA
INSEGNAMENTO	CALCOLATORI ELETTRONICI C.I.
CODICE INSEGNAMENTO	18073
MODULI	Si
NUMERO DI MODULI	3
SETTORI SCIENTIFICO-DISCIPLINARI	ING-INF/05
DOCENTE RESPONSABILE	GAGLIO SALVATORE Professore Ordinario Univ. di PALERMO
ALTRI DOCENTI	GAGLIO SALVATORE Professore Ordinario Univ. di PALERMO ORTOLANI MARCO Ricercatore Univ. di PALERMO PERI DANIELE Ricercatore Univ. di PALERMO
CFU	18
PROPEDEUTICITA'	
MUTUAZIONI	
ANNO DI CORSO	1
PERIODO DELLE LEZIONI	Annuale
MODALITA' DI FREQUENZA	Facoltativa
TIPO DI VALUTAZIONE	Voto in trentesimi
ORARIO DI RICEVIMENTO DEGLI STUDENTI	ORTOLANI MARCO Lunedì 15:00 17:00 Ufficio docente PERI DANIELE Mercoledì 15:00 16:00 Ricevimento in modalita a distanza sulla piattaforma MS Teams

DOCENTE: Prof. SALVATORE GAGLIO

PREREQUISITI	Nessuno
RISULTATI DI APPRENDIMENTO ATTESI	<p>Conoscenza e capacita di comprensione</p> <p>Lo studente acquisira' una conoscenza sia teorica che di tipo progettuale riguardante le reti logiche, la programmazione strutturata in linguaggio C e le architetture di base dei moderni calcolatori elettronici. Conoscera' i principali strumenti di programmazione, gli elementi di rappresentazione delle informazioni nei calcolatori, le metodologie di base per la progettazione e l'analisi di reti logiche combinatorie e sequenziali, gli aspetti di base delle architetture dei calcolatori e dei sistemi operativi Unix-like.</p> <p>Capacita di applicare conoscenza e comprensione</p> <p>Lo studente sara' in grado di valutare le possibili soluzioni software a problemi di complessita' media e affrontarne l'implementazione utilizzando strumenti e ambienti di sviluppo per la programmazione in linguaggio C in ambienti Unix-like. Sara' in grado di affrontare semplici problemi di rappresentazione binaria delle informazioni e sara' anche in grado di progettare a livello funzionale circuiti logici per la soluzione di semplici problemi.</p> <p>Autonomia di giudizio</p> <p>Lo studente sara' in grado di affrontare in autonomia l'analisi, la progettazione e l'implementazione di moduli digitali e di software basato sulla programmazione strutturata. Sara' in grado di valutare la qualita' delle sue soluzioni in termini di semplicita, leggibilita, strutturazione, efficienza e riutilizzabilita.</p> <p>Abilita comunicative</p> <p>Lo studente sara' in grado di esporre, efficacemente e con proprieta' di linguaggio, analisi e soluzioni di problemi affrontabili con la programmazione strutturata e con la progettazione funzionale di circuiti logici, nonche' di problemi di rappresentazione delle informazioni.</p> <p>Capacita d'apprendimento</p> <p>Lo studente sara' in grado di affrontare in maniera autonoma problemi di progettazione di circuiti digitali e di programmazione strutturata individuando e integrando soluzioni parziali gia' disponibili, sia formalizzate sia implementate. Sara' in grado di approfondire in autonomia la conoscenza di moduli hardware e software e di interfacce di programmazione. Sara in grado di approfondire la conoscenza dei linguaggi e dei paradigmi di programmazione, dei sistemi operativi, delle architetture dei calcolatori e dei circuiti logici.</p>
VALUTAZIONE DELL'APPRENDIMENTO	<p>Le conoscenze e le competenze acquisite dallo studente saranno verificate attraverso una prova finale scritta. La prova finale puo' essere sostituita da diverse prove scritte parziali nel corso dell'anno. Ogni prova scritta riguardera' degli esercizi che verteranno sui contenuti dei moduli dell'insegnamento. Essa ha lo scopo di verificare le conoscenze dello studente riguardo agli argomenti affrontati durante il corso e la sua capacita' di applicare le conoscenze acquisite. La valutazione della prova terra' conto della capacita' di comunicazione delle scelte implementative da parte dello studente, nonche' la sua autonomia di giudizio nel proporre soluzioni adeguate all'applicazione richiesta. La valutazione avviene in trentesimi.</p>
ORGANIZZAZIONE DELLA DIDATTICA	<p>Lezioni frontali Esercitazioni in aula e nelle aule informatiche</p>

**MODULO
ARCHITETTURE DI BASE DEI CALCOLATORI**

Prof. DANIELE PERI

TESTI CONSIGLIATI

1. Andrew S. Tanenbaum, Architettura dei calcolatori un approccio strutturale, quinta edizione, Pearson-Prentice Hall

TIPO DI ATTIVITA'	A
AMBITO	50283-Matematica, informatica e statistica
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	96
NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE	54

OBIETTIVI FORMATIVI DEL MODULO

Al termine del corso lo studente conoscerà i concetti necessari alla comprensione della struttura dei calcolatori elettronici e la loro evoluzione storica. Concetti di base necessari alla comprensione della struttura dei calcolatori elettronici digitali programmabili. Imparerà ad usare il linguaggio assembleativo per la programmazione della architettura simulata Mic-1/IJVM.

PROGRAMMA

ORE	Lezioni
4	Calcolatori elettronici digitali programmabili e loro evoluzione storica. Struttura di un calcolatore. Modello Von Neumann: CPU, memoria, input/output. Microprocessori.
4	Rappresentazione delle informazioni. Rappresentazione di interi con segno. Rappresentazione in complemento alla base. Rappresentazione di numeri reali: virgola fissa, virgola mobile. Rappresentazione di caratteri alfanumerici. Codice ASCII.
4	Il calcolatore elettronico: l'architettura di Von Neumann. Organizzazione a livelli di un sistema di calcolo. Riepilogo sul livello logico digitale.
4	Il livello di microarchitettura. Il ciclo fetch-decode-execute. Il percorso dati. Il concetto di microistruzione. Funzione ed uso dei registri.
4	Componenti di base della microarchitettura Mic-1: percorso dati, registri e ALU. Unita' di controllo del Mic-1. Processo di esecuzione delle istruzioni.
4	Architettura del Mic-1 e IJVM. l'Instruction Set Architecture. Istruzioni calcolo e di movimento dati. Istruzioni di controllo. Rivisitazione del data path.
4	Introduzione alla programmazione di basso livello. Il problem solving: decomposizione sistematica. I costrutti di base: sequenze, condizioni e iterazioni. Istruzioni ed implementazione di costrutti di controllo di flusso in IJVM su Mic-1. Il linguaggio assembleativo. Lo stack. Struttura di base. Ingresso/uscita con interrupt. Aritmetica attraverso lo stack e conversione dei dati.
ORE	Esercitazioni
12	Implementazione di costrutti di alto livello in assembly
12	Uso di un simulatore per il livello ISA IJVM e corrispondenza con il livello di microarchitettura Mic-1

**MODULO
RETI LOGICHE**

Prof. MARCO ORTOLANI

TESTI CONSIGLIATI

M. Morris Mano, Charles R. Kime, "Reti Logiche", Ed. Italiana, Pearson Education

TIPO DI ATTIVITA'	A
AMBITO	50283-Matematica, informatica e statistica
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	96
NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE	54

OBIETTIVI FORMATIVI DEL MODULO

Al termine del corso lo studente conoscerà i concetti necessari alla comprensione della struttura dei calcolatori elettronici e la loro evoluzione storica. Conoscerà le principali nozioni sull'algebra di Boole e sulle reti logiche. Avrà conoscenza delle problematiche inerenti le metodologie di progettazione di reti logiche combinatorie e sequenziali.

PROGRAMMA

ORE	Lezioni
2	Calcolatori elettronici digitali programmabili e loro evoluzione storica. Struttura di un calcolatore. Modello Von Neumann: CPU, memoria, input/output. Microprocessori.
4	Rappresentazione delle informazioni. Rappresentazione interna ed esterna. Sistemi di numerazione. Numerazione binaria. Bit, byte e multipli. Conversioni binario-decimale. Operazioni aritmetiche. Sistema di numerazione ottale. Sistema di numerazione esadecimale. Codici. Codice BCD.
6	Algebra Booleana Operatori e porte logiche. Funzioni. Tabelle di verità. Diagrammi e circuiti logici. Identità fondamentali. Principio di dualità. Teorema di de Morgan. Complemento di una funzione. Forme canoniche. Mintermini. Maxtermini
5	Sintesi a due livelli. Mappe di Karnaugh. Implicanti, primi implicanti e primi implicanti essenziali di funzioni booleane. Minimizzazione di funzioni booleane. Operatore XOR. Operatori funzionalmente completi.
5	Reti combinatorie Decoder ed encoder. Espansione in serie di decoder. Encoder con priorità. Multiplexer e demultiplexer. Sintesi con decoder. Sintesi con multiplexer. Sommatori.
8	Reti sequenziali Modelli di Mealy e Moore. Latch. Flip-Flop. Analisi di reti sequenziali sincrone. Sintesi di reti sequenziali sincrone.
5	Memorie e dispositivi logici programmabili Generalità sui dispositivi di memoria e sui dispositivi logici programmabili. Tecnologie logiche programmabili. Memorie a sola lettura (ROM). Implementazione di circuiti combinatori mediante ROM. Dispositivi a matrice logica programmabile (PAL). Dispositivi logici a matrice programmabile (PLA). Cenni sui dispositivi logici programmabili VLSI e FPGA.

ORE	Esercitazioni
6	Progettazione, sintesi e simulazione di circuiti combinatori di base.
6	Progettazione, sintesi e simulazione di semplici circuiti sequenziali.

**MODULO
FONDAMENTI DI PROGRAMMAZIONE**

Prof. SALVATORE GAGLIO

TESTI CONSIGLIATI

P. Deitel, H. Deitel, "Il Linguaggio C – Fondamenti e tecniche di programmazione", Ottava edizione, Pearson Italia, 2016.

TIPO DI ATTIVITA'	A
AMBITO	50283-Matematica, informatica e statistica
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	96
NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE	54

OBIETTIVI FORMATIVI DEL MODULO

Il modulo si propone di fornire allo studente i concetti di base nell'ambito della programmazione dei calcolatori elettronici. Durante il corso vengono affrontate le tecniche di sviluppo di programmi per affinamenti successivi secondo la tecnica della programmazione strutturata con l'uso del linguaggio C, con lo scopo di realizzare applicazioni concrete. L'approccio sarà orientato alla costruzione di algoritmi e alla strutturazione e gestione dei dati.

PROGRAMMA

ORE	Lezioni
2	Introduzione al modulo. Introduzione alla programmazione.
2	Introduzione alla programmazione nel linguaggio C: Aritmetica e decisione.
2	Sviluppo di un programma strutturato in C.
2	Controllo nei programmi in C.
4	Funzioni in C e ricorsione.
4	Array in C, ricerca e ordinamento.
4	Puntatori in C.
2	Strutture, unioni, manipolazione di bit ed enumerazioni in C.
2	Elaborazione di file in C.
4	Strutture di dati in C.

ORE	Esercitazioni
2	Introduzione all'uso della linea dei comandi e agli ambienti di sviluppo.
8	Sviluppo di programmi strutturati in C.
6	Sviluppo di programmi in C per la manipolazione di vettori e matrici.
4	Sviluppo di programmi in C che fanno uso di puntatori.
4	Sviluppo di programmi in C per la gestione di strutture di dati.