



# UNIVERSITÀ DEGLI STUDI DI PALERMO

<b>DIPARTIMENTO</b>	Ingegneria		
<b>ANNO ACCADEMICO OFFERTA</b>	2023/2024		
<b>ANNO ACCADEMICO EROGAZIONE</b>	2023/2024		
<b>CORSO DILAUREA</b>	INGEGNERIA ROBOTICA		
<b>INSEGNAMENTO</b>	CALCOLATORI ELETTRONICI C.I.		
<b>CODICE INSEGNAMENTO</b>	18794		
<b>MODULI</b>	Si		
<b>NUMERO DI MODULI</b>	2		
<b>SETTORI SCIENTIFICO-DISCIPLINARI</b>	ING-INF/05		
<b>DOCENTE RESPONSABILE</b>	CHELLA ANTONIO	Professore Ordinario	Univ. di PALERMO
<b>ALTRI DOCENTI</b>	PIRRONE ROBERTO	Professore Ordinario	Univ. di PALERMO
	CHELLA ANTONIO	Professore Ordinario	Univ. di PALERMO
<b>CFU</b>	12		
<b>PROPEDEUTICITA'</b>			
<b>MUTUAZIONI</b>			
<b>ANNO DI CORSO</b>	1		
<b>PERIODO DELLE LEZIONI</b>	Annuale		
<b>MODALITA' DI FREQUENZA</b>	Facoltativa		
<b>TIPO DI VALUTAZIONE</b>	Voto in trentesimi		
<b>ORARIO DI RICEVIMENTO DEGLI STUDENTI</b>	<b>CHELLA ANTONIO</b> Lunedì 09:00 11:00 DICGIM, edificio 6, III piano <b>PIRRONE ROBERTO</b> Mercoledì 11:30 13:00 Studio del docente, Edificio 6, terzo piano, stanza 3025		

<p><b>PREREQUISITI</b></p>	<p>Conoscenze di base di matematica e scienze apprese alla scuola media superiore.</p>
<p><b>RISULTATI DI APPRENDIMENTO ATTESI</b></p>	<p>Conoscenza e capacita' di comprensione          Lo studente alla fine del corso acquisira' una buona conoscenza delle tecniche di rappresentazione delle informazione e della loro elaborazione tramite reti sequenziali e combinatorie, e una buona conoscenza delle tecniche di ottimizzazione di tali sistemi.          Lo studente apprendera' la sintassi e le tecniche di programmazione in linguaggio C, le strutture dati e gli algoritmi fondamentali. Infine lo studente acquisira' una conoscenza basilare della programmazione dei robot in ambiente simulato usando il linguaggio C.</p> <p>Capacita' di applicare conoscenza e comprensione          Lo studente sara' in grado di applicare le tecniche studiate per progettare, a livello logico e funzionale, reti combinatorie e sequenziali e analizzarne il funzionamento. Lo studente sarà in grado di analizzare semplici casi d'uso relativi all'applicazione di reti combinatorie e sequenziali per il controllo di robot reattivi.          Lo studente sara' in grado di utilizzare strumenti e ambienti di sviluppo per la programmazione in linguaggio C e sapra' implementare soluzioni software. Lo studente sara' in grado di concepire e scrivere semplici programmi di controllo dei robot in linguaggio C all'interno del simulatore WeBots.</p> <p>Autonomia di giudizio          Lo studente sara' in grado sia di effettuare l'analisi di un problema che di progettare, a partire da una descrizione verbale, una opportuna soluzione software. Sara' in grado di valutare la qualita' di una soluzione software in termini di semplicita', leggibilita', efficienza e possibilita' di riutilizzo. Sara' in grado di capire i principi di funzionamento del calcolatore.</p> <p>Abilita' comunicative          Lo studente acquisira' la capacita' di comunicare ed esprimere problematiche inerenti l'oggetto del corso. Sara' in grado di descrivere con terminologia appropriata un circuito logico. Sara' in grado, utilizzando un linguaggio semplice e chiaro, di descrivere i processi di analisi e di sintesi di soluzioni software.</p>
<p><b>VALUTAZIONE DELL'APPRENDIMENTO</b></p>	<p>L'apprendimento viene valutato attraverso due prove scritte in itinere che insistono sui due moduli del corso ed una prova orale opzionale. Lo studente che non partecipa alle prove in itinere dovrà fare una prova scritta finale sui contenuti dei due moduli.</p> <p>La prova scritta riguardante gli argomenti del modulo di Reti Logiche consiste prevalentemente nella risposta ad alcuni quesiti a risposta chiusa o aperta e in quesiti di progettazione di circuiti sequenziali e combinatori.          I quesiti tenderanno a verificare a) le conoscenze acquisite; b) la capacita' di progettare circuiti che rispondano ad alcune specifiche, c) la capacita' di organizzare ed esporre le conoscenze tecniche.          Il punteggio massimo si ottiene quando la verifica accerta il pieno possesso dei tre seguenti aspetti: capacita' di svolgere i quesiti riguardanti la rappresentazione dell'informazione, la capacita' di progettare o ottimizzare circuiti logici che assolvono a specifici compiti, la capacita' di descrivere e confrontare diverse soluzioni circuitali.</p> <p>La prova scritta riguardante gli argomenti del modulo di Fondamenti di Programmazione viene effettuata al computer. La prova consiste nella generazione di uno o più programmi in linguaggio C in base ad alcune specifiche tecniche date.          La prova tendera' a verificare a) la conoscenza del linguaggio di programmazione C; b) la capacita' di trovare soluzioni a semplici problemi tipici della disciplina; c) la capacita' di creare software funzionanti.          Il punteggio massimo si ottiene se la verifica accerta il pieno possesso dei tre seguenti aspetti: l'utilizzo di costrutti del linguaggio in modo sintatticamente corretto; la capacita' di comporre i costrutti per risolvere problemi riguardanti l'acquisizione, l'elaborazione e la memorizzazione di informazioni; la creazione di software che possano essere eseguiti senza presentare malfunzionamenti.          Il voto finale sarà calcolato come media delle valutazioni delle 2 prove.          La prova orale, cui lo studente potrà accedere facoltativamente solo previa indicazione dei docenti riguardera' gli argomenti di entrambi i moduli e consisterà di una discussione generale su tutti gli aspetti delle Reti Logiche e della Programmazione in Linguaggio C.</p> <p>La valutazione generale si basa sui seguenti criteri di massima:          a) eccellente (30 - 30 e lode): ottima conoscenza degli argomenti, ottima proprieta' di linguaggio, buona capacita' analitica, lo studente e' in grado di applicare le conoscenze per risolvere i problemi proposti;          b) molto buono (26 - 29): buona padronanza degli argomenti, piena proprieta' di linguaggio, lo studente e' in grado di applicare le conoscenze per risolvere i</p>

	<p>problemi proposti;</p> <p>c) buono (24 - 25): conoscenza di base dei principali argomenti, discreta proprietà di linguaggio con limitata capacità di applicare autonomamente le conoscenze alla soluzione dei problemi proposti;</p> <p>d) discreto (21 - 23): limitata padronanza degli argomenti principali dell'insegnamento, discreta proprietà di linguaggio, discreta capacità di applicare autonomamente le conoscenze acquisite;</p> <p>e) sufficiente (18 - 20): minima conoscenza di base degli argomenti principali dell'insegnamento e del linguaggio tecnico, scarsa capacità di applicare autonomamente le conoscenze acquisite</p> <p>f) insufficiente: non possiede una conoscenza minima accettabile dei contenuti degli argomenti trattati nell'insegnamento.</p>
<b>ORGANIZZAZIONE DELLA DIDATTICA</b>	Lezioni frontali, esercitazioni.

<p><b>MODULO</b></p> <p><b>ARCHITETTURE DEI CALCOLATORI</b></p> <p><i>Prof. ANTONIO CHELLA</i></p>	
<b>TESTI CONSIGLIATI</b>	
Mano M. Morris, Charles R. Kime, Tom Martin, Reti Logiche - quinta edizione, Pearson Education Italia (2019), ISBN: 889190581X	
<b>TIPO DI ATTIVITA'</b>	A
<b>AMBITO</b>	50283-Matematica, informatica e statistica
<b>NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE</b>	96
<b>NUMERO DI ORE RISERVATE ALLE ATTIVITA' DIDATTICHE ASSISTITE</b>	54
<b>OBIETTIVI FORMATIVI DEL MODULO</b>	
Lo studente sarà capace di applicare le metodologie studiate in contesti differenti e di apprendere processi di analisi e sintesi relativi ai circuiti logici, anche applicati a semplici problemi robotici.	

### PROGRAMMA

ORE	Lezioni
6	Introduzione al corso, concetto di informazione e sua natura logaritmica. Rappresentazione dei numeri interi in una base generica, numeri binari. Conversione di numeri da una base ad un'altra, rappresentazione di numeri in virgola mobile e di numeri negativi. Rappresentazione numerica di immagini e suoni, codice ASCII.
6	Introduzione all'algebra di Boole. Porte logiche, funzioni booleane, logica negata, teoremi e proprietà dell'algebra, forme normali. Mappe di Karnaugh, minimizzazione di funzione booleane. Sintesi di reti combinatorie. Circuiti integrati (full adder, sommatore, multiplexer, decoder).
4	Circuiti sequenziali. Modelli di Mealy e di Moore. Latch SR, Latch di tipo D, flip flop JK, T, D, SR. Master slave, flip flop sensibili alle variazioni. Analisi di reti sequenziali sincrone. Concetto di stato, diagramma di stato. Equazioni caratteristiche dei flip-flop.
4	Procedimenti di sintesi delle reti sequenziali sincrone: codifica degli stati. Sintesi di reti sequenziale. Sintesi con flip-flop D. Verifica del funzionamento di reti sequenziali, Simulazione di reti sequenziali; Sintesi con flip-flop D, flip-flop T e flip-flop JK.
4	Sintesi di riconoscitori di sequenza, Sintesi di reti sequenziali a partire dal diagramma di stato. Registri, Contatori, ROM, PAL, PLA.
6	Architettura di un calcolatore, CPU, ALU, RAM, BUS, macchina di von Neumann. Concetto di programma. Processore. Sistemi CISC, sistemi RISC. Logica cablata, logica micro-programmata.
6	Applicazioni e casi di studio al controllo di semplici sistemi robotici.

ORE	Esercitazioni
3	Rappresentazione dell'informazione. Algebra Booleana.
3	Analisi e Sintesi di reti Combinatorie, Rappresentazione sulla Mappa di Karnaugh; Forme Canoniche Congiuntive e Disgiuntive.
3	Analisi e Sintesi di Reti Sequenziali, Riconoscitori di Sequenze.
3	Casi d'uso di architetture dei calcolatori
3	Problematiche dell'analisi e sintesi di reti combinatorie e sequenziali per il controllo di semplici robot.
3	Analisi e sintesi di reti combinatorie e sequenziali per il controllo di semplici robot: casi d'uso

**MODULO  
FONDAMENTI DI PROGRAMMAZIONE**

*Prof. ROBERTO PIRRONE*

**TESTI CONSIGLIATI**

1. Jeri R. Anly – Elliot B. Koffman, Problem solving e programmazione in C, Apogeo, ISBN-10: 8838786410  
2. J. Glenn Brookshear - Stephen G. Kochan, Fondamenti di informatica e programmazione in C, Pearson, ISBN-10: 8865183691

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

**OBIETTIVI FORMATIVI DEL MODULO**

Al termine del corso lo studente sarà in grado di valutare, analizzare, comunicare e implementare le possibili soluzioni software a semplici problemi utilizzando l'acquisita padronanza del linguaggio C. Egli avrà inoltre contezza del simulatore robotico WeBots e della sua programmazione in linguaggio C.

**PROGRAMMA**

ORE	Lezioni
2	Introduzione al corso. Elaborazione dei dati, architettura dei computer, sistemi operativi, reti di computer. Linguaggi di programmazione, prospettiva storica. Traduzione dei programmi. Indipendenza dalla macchina
2	Concetto di algoritmo, rappresentazione degli algoritmi, pseudocodice e diagrammi di flusso. Concetti della programmazione tradizionale. Compilazione del primo programma. Esecuzione del primo programma.
2	Introduzione al linguaggio C. Variabili, tipi di dati ed espressioni aritmetiche. Differenza tra variabili e costanti. Assegnamento. Operatori. Priorità tra gli operatori.
2	Concetto di sottoprogramma. Introduzione alle funzioni C. Valori di ritorno delle funzioni. Funzioni con parametri e senza parametri. Progettazione modulare del software: tecniche top-down e diagrammi di struttura. Comprensione e simulazione di algoritmi/programmi.
2	Strutture di selezione (if-else) e di selezione tra più alternative (switch-case). Diagrammi di flusso per le strutture di selezione. Comprensione e simulazione di algoritmi/programmi.
2	Programmi iterativi in linguaggio C. Strutture iterative (for, while e do-while). Diagrammi di flusso per le strutture di iterative. Annidamento delle strutture. Comprensione e simulazione di algoritmi/programmi.
2	Algoritmi iterativi. Efficienza e correttezza di un algoritmo. Introduzione all'analisi computazionale di un algoritmo: complessità di tempo e di memoria.
4	Approfondimento sulle funzioni. Variabili locali. Visibilità e scope di una variabile. Variabili globali, static e register. Concetto di puntatore e suo uso nei parametri di funzione.
2	Concetto di ricorsione. Gestione della memoria di una funzione ricorsiva. Implementazione in linguaggio C.
3	Concetto di struttura di dati. Gli array: dichiarazione e indicizzazione. Array multidimensionali. Tipo enumerativo e array con indice di tipo enumerativo. Cicli e array. Semplici strutture dati con array: le pile.
2	Stringhe di caratteri. Utilizzo delle stringhe in C: array di caratteri, stringhe di caratteri di lunghezza variabile, sequenze di escape.
4	Approfondimento sui puntatori: puntatori e array. Aritmetica dei puntatori. Strutture e unioni. Implementazione di strutture e tipi personalizzati. Implementazione di strutture dati elementari: liste e code. Operazioni di ricerca e di ordinamento in collezioni di dati.
2	Gestione di input e output nel linguaggio C. Gestione di file.
2	Direttive di preprocessore. Inclusione di librerie personali.
ORE	Esercitazioni
4	Progettazione di semplici algoritmi attraverso diagrammi di flusso. Passaggio da diagrammi di flusso a codice C. Primi programmi in C: compilazione, linking ed esecuzione. Implementazione di programmi per la manipolazione di dati numerici.
2	Progettazione di algoritmi che prevedono la selezione e strutture iterative attraverso i diagrammi di flusso. Implementazione corrispondente in linguaggio C.
3	Esercizi sulla progettazione di software modulare attraverso un uso corretto delle funzioni. Progettazione e implementazione di algoritmi che utilizzano la ricorsione.
6	Progettazione e implementazione di algoritmi che utilizzano stringhe di caratteri, vettori e matrici. Utilizzo dei puntatori. Esercizi sull'implementazione di liste, code e pile.
3	Implementazione di programmi che usano l'I/O da file. Preparazione alla prova scritta.

