# FISICA (LB23)

(Lecce - Università degli Studi)

## Insegnamento INFORMATICA

GenCod A004601

**Docente titolare** Rosella CATALDO

**Insegnamento** INFORMATICA

**Anno di corso** 1

**Insegnamento in inglese** COMPUTER SCIENCE

**Lingua** ITALIANO

**Settore disciplinare** INF/01

**Percorso** PERCORSO COMUNE

**Corso di studi di riferimento** FISICA

**Tipo corso di studi** Laurea

**Sede** Lecce

**Crediti** 6.0

**Periodo** Secondo Semestre

**Ripartizione oraria** Ore Attività frontale: 52.0

**Tipo esame** Orale

**Per immatricolati nel** 2019/2020

**Valutazione** Voto Finale

**Erogato nel** 2019/2020

**Orario dell'insegnamento** https://easyroom.unisalento.it/Orario

BREVE DESCRIZIONE DEL CORSO

Il Corso parte dall'introduzione dei princìpi di base della materia, per poi svilupparsi nell'analisi delle metodologie e tecnologie per le reti locali e geografiche.
I concetti teorici del linguaggio di programmazione C saranno integrati e applicati con esemplificazioni a problemi di interesse nel particolare corso di Laurea. Verranno illustrati i concetti inerenti alla strutturazione di semplici classi in C++, che saranno poi sviluppate, mediante utilizzo di apposito software.

PREREQUISITI

**Le nozioni matematiche di base acquisite negli studi superiori.**

OBIETTIVI FORMATIVI

**Conoscenze e comprensione**: Possedere una solida preparazione con un ampio spettro di conoscenze di base di tipo informatico.
**Capacità di applicare conoscenze e comprensione**: Fornire ai destinatari una base di conoscenze attinenti agli argomenti propri del percorso formativo,
- utilizzare strumenti di calcolo matematico e tecnologie informatiche utili al completamento del corso di Laurea nel suo complesso,
- essere in grado di formalizzare algoritmi di risoluzione di problemi di moderata difficoltà, in modo da facilitare la loro analisi e implementazione,
- essere capaci di leggere e comprendere, in modo autonomo, testi di base di Informatica e programmi di moderata difficoltà in C e C++.
**Autonomia di giudizio**: L'esposizione dei contenuti e delle argomentazioni sarà svolta in modo da migliorare la capacità dello studente
di riconoscere le problematiche inerenti all'architettura dei calcolatori e alla formalizzazione di aspetti computazionali.
**Abilità comunicative:** La presentazione degli argomenti sarà svolta in modo da consentire l'acquisizione di una buona capacità di comunicare problemi, idee e soluzioni riguardanti l'Informatica, sia come comprensione dell tematiche fondamentali inerenti alla Scienza dei Calcolatori che come sviluppo di codice in un liguaggio di programmazione (C).
**Capacità di apprendimento:** Saranno indicati argomenti da approfondire, strettamente correlati con l'insegnamento, al fine di stimolare la capacità di apprendimento autonomo dello studente.

METODI DIDATTICI

Le lezioni si svolgono negli orari previsti dal calendario del Corso di Laurea, con modalità frontale. Sono contemplate esercitazioni assistite, all'interno dell'orario delle lezioni. Durante le esercitazioni sia in aula che in Laboratorio, lo studente dovrà cimentarsi nell'implementazione degli algoritmi proposti a lezione, dei quali sarà sempre fornita una falsariga di sviluppo.

MODALITA' D'ESAME

La valutazione finale, espressa in trentesimi, sarà conseguita sulla base della qualità dell'elaborato oggetto della prova d'esame, concernente i contenuti del corso.
L'esame si articola in:

- uno scritto, relativo all'architettura del computer,
- un progetto in C, da sviluppare con utilizzo di Dev C++ o CodeBlocks,
- strutturazione di semplici classi in progetti C++ , da sviluppare con utilizzo di Dev C++ o CodeBlocks,
- elaborazione del makefile UNIX, relativo a un progetto C assegnato, e strutturazione di uno script in awk (riservato agli studenti fino agli anni accademici 2016-2017).

Un eventuale colloquio, se ritenuto necessario dal docente.

PROGRAMMA ESTESO

Architettura generale del calcolatore. Dalla macchina reale alla macchina virtuale. Introduzione alla codifica dell' informazione e alle problematiche inerenti ai metodi di codifica. Codifiche numeriche e cambiamenti di base in sistemi posizionali.Codifica in complemento alla base e alla base diminuita.Codifiche numeriche in formato floating point; codifica dei caratteri. Conversioni di segnale analogico in segnale digitale. Tecniche di compressione dei dati. Codifica di Huffman. Variabili e tipi di variabili. Il concetto di algoritmo. Strutturazione di algoritmi. Algoritmi di ordinamento. Ricerca binaria e ricerca sequenziale. Il processore. Interazione del processore con la Memoria principale. Dai processori CISC ai RISC. Caratteristiche prestazionali di un processore. Le memorie del computer: caratteristiche e utilizzo. Il software di base: BIOS e Sistema Operativo. Dal modello monolitico al Client- server. Definizione e funzioni del Sistema Operativo. Il gestore dei processi, del processore e della memoria. Gestione dell'I/O. Reti di computer. Il modello ISO/OSI e lo standard TCP/IP. Il modello client-server. Servizi di rete. Tipologie di rete e tecniche di trasmissione. Il DNS. Introduzione al linguaggio C. Le variabili: definizione in tipo e riempimento. Controllo di flusso: if, while, for, case. Generazione di numeri random. I vettori e le matrici. Il concetto di funzione in C. Dal programma al progetto. Esempi di strutturazione e di utilizzo. I puntatori. Utilizzo di puntatori in funzioni. I puntatori a file. Lettura e scrittura su file binari.
Introduzione alla programmazione strutturata in C++. Conversione di progetti C in progetti C++. Il concetto di classe: definizione dell'interfaccia della classe, proprietà e metodi degli oggetti.. Sviluppo di progetti in C++ con allocazione dinamica di memoria. Esempio di scambio di dati fra classi non derivate fra loro. Realizzazione della classe Coppia. Un esempio di overloading. Sviluppo di alcune semplici classi, da inserire in appositi progetti.

TESTI DI RIFERIMENTO

- **Paolo Tosoratti: Introduzione all'Informatica. CEA**
- **Ceri, Mandrioli, Sbattella. Informatica: arte e mestiere. McGraw-Hill, 2008.**
- **Kernighan & Ritchie. The C Programming Language. Prentice Hall. · Ellie Quingley. UNIX shells by example. Upper Saddle River, NJ : Prentice Hall Professional Technical Reference.**

Eventuali link ad altro materiale verrà messo a disposizione durante il corso, sulle pagine WEB del docente.