

# MATEMATICA (LB04)

(Lecce - Università degli Studi)

## Insegnamento ALGORITMI E STRUTTURE DATI

GenCod A003874

Docente titolare VITTORIO BILO'

**Insegnamento** ALGORITMI E STRUTTURE DATI

**Insegnamento in inglese** ALGORITHMS AND DATA STRUCTURES

**Settore disciplinare** INF/01

**Corso di studi di riferimento** MATEMATICA

**Tipo corso di studi** Laurea

**Crediti** 6.0

**Ripartizione oraria** Ore Attività frontale: 42.0

**Per immatricolati nel** 2019/2020

**Erogato nel** 2021/2022

**Anno di corso** 3

**Lingua** ITALIANO

**Percorso** PERCORSO COMUNE

**Sede** Lecce

**Periodo** Primo Semestre

**Tipo esame** Scritto

**Valutazione** Voto Finale

**Orario dell'insegnamento**

<https://easyroom.unisalento.it/Orario>

### BREVE DESCRIZIONE DEL CORSO

Il corso di Algoritmi e Strutture Dati è rivolto a quegli studenti che, avendo già acquisito una buona padronanza di un linguaggio di programmazione, vogliono espandere le loro conoscenze sulla progettazione avanzata di soluzioni algoritmiche per problemi computazionali.

### PREREQUISITI

Si richiede che lo studente abbia ben metabolizzato un linguaggio di programmazione e i concetti di astrazione funzionale e programmazione ricorsiva; e che sia in grado di dar vita a dei, sia pur semplici, processi di problem solving. E' inoltre auspicabile una certa familiarità con discipline formali come l'algebra, la geometria, l'analisi matematica e il calcolo delle probabilità e, in particolar modo, con argomenti come la logica proposizionale, le dimostrazioni per induzione e per assurdo, gli insiemi numerabili, le matrici, il valore atteso di una variabile aleatoria.

### OBIETTIVI FORMATIVI

Conoscenze e comprensione: conoscere gli strumenti teorici alla base dell'analisi e progettazione di algoritmi.

Capacità di applicare conoscenze e comprensione: essere in grado di progettare algoritmi efficienti per problemi computazionali avanzati.

Autonomia di giudizio: essere in grado di valutare, tra le molteplici soluzioni possibili di un dato problema, quelle migliori o che meglio soddisfino certi requisiti.

Abilità comunicative: saranno illustrati strumenti teorici atti a comprendere e comunicare problematiche, modelli e soluzioni tipici dell'area della Teoria degli Algoritmi e della Complessità Computazionale.

Capacità di apprendimento: gli studenti saranno stimolati a implementare le soluzioni proposte durante le lezioni.

### METODI DIDATTICI

Lezioni teoriche frontali corredate da vari esercizi.

### MODALITA' D'ESAME

Prova scritta volta ad accertare non solo la conoscenza degli strumenti teorici illustrati durante il corso, ma anche la capacità del candidato di risolvere, in maniera efficiente, problemi computazionali.

---

## PROGRAMMA ESTESO

Problemi Computazionali: problemi decidibili e indecidibili, problemi trattabili e intrattabili.  
Complessità di un Algoritmo: il modello RAM a costi uniformi, notazione asintotica, le classi di complessità P, NP, EXP (cenni), problemi NP-completi (cenni).  
Esempi di Analisi della Complessità: algoritmi Selection Sort e Insertion Sort.  
Limitazioni Superiori e Inferiori alla Complessità di un Problema Computazionale: algoritmi ottimi.  
Esempi di Algoritmi Ottimi: calcolo del segmento di somma massima, ricerca sequenziale e binaria di una chiave in un array.  
Paradigma Divide et Impera: definizione della tecnica.  
Metodi di Soluzione di Relazioni di Ricorrenza: metodo di iterazione, metodo di sostituzione, metodo dell'albero di ricorsione, metodo del cambio di variabile, Teorema Master e sua dimostrazione.  
Algoritmi Divide et Impera: moltiplicazione di interi di lunghezza arbitraria, moltiplicazioni di matrici, Merge Sort, Quick Sort.  
Paradigma della Programmazione Dinamica: definizione della tecnica.  
Algoritmi di Programmazione Dinamica: parentesizzazione ottima del prodotto di n matrici, sottosequenza comune di lunghezza massima, partizionamento, problema dello zaino.  
Algoritmi Pseudo-Polinomiali.  
Le liste.  
Il Problema dei Matrimoni Stabili.  
Alberi binari: visite, problemi decomponibili, soluzione efficiente in spazio al problema del minimo antenato comune.  
Il Problema del Dizionario: implementazione tramite liste doppie, tabelle hash con liste di trabocco e a indirizzamento aperto, alberi binari di ricerca e alberi AVL (cenni).

---

## TESTI DI RIFERIMENTO

Pierluigi Crescenzi, Giorgio Gambosi, Roberto Grossi, "Strutture di Dati e Algoritmi", Pearson Editore.