

Algoritmi e Strutture di Dati

Introduzione al corso

Gianluigi Zavattaro
Dip. di Informatica – Scienza e Ingegneria
Università di Bologna
gianluigi.zavattaro@unibo.it

Pietro Di Lena
Dip. di Informatica – Scienza e Ingegneria
Università di Bologna
pietro.dilena@unibo.it

Slide realizzate a partire da materiale fornito dal Prof. Moreno Marzolla



Copyright © 2010, Moreno Marzolla, Università di Bologna, Italy
(<http://www.moreno.marzolla.name/teaching/ASD2010/>)

This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Presentiamoci

- Gianluigi Zavattaro: gianluigi.zavattaro@unibo.it
Pietro Di Lena: pietro.dilena@unibo.it
- Lezioni
 - Lunedì ore 16:00 – 18:00, aula Cremona, Dip. Matematica
 - Martedì ore 11:00 – 14:00, aula M1, Dip. Mineralogia
 - Giovedì ore 9:00 – 11:00, aula M2, Dip. Mineralogia
 - Venerdì ore 9:00 – 12:00, aula M2, Dip. Mineralogia
- Orario di ricevimento Gianluigi Zavattaro
 - Da richiedere via e-mail
 - Ufficio al piano terra della palazzina in Mura A. Zamboni 7
- Orario di ricevimento Pietro Di Lena
 - Da richiedere via e-mail
 - Ufficio al piano terra della palazzina in via Malaguti 1/d

Informazioni generali sul corso

Sito web del corso

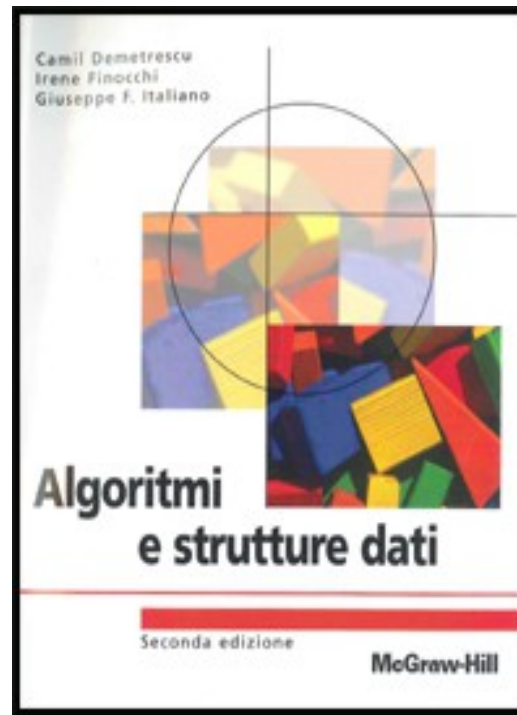
- <https://virtuale.unibo.it/course/view.php?id=27602>



- Avvisi (comunicazioni verranno inviate tramite la piattaforma)
- Lucidi delle lezioni e esempi di codice Java
- Agenda con contenuti delle lezioni
- Esercizi ed esempi di esami
- Specifiche e consegna dei progetti

Testo adottato

- Camil Demetrescu, Irene Finocchi,
Giuseppe F. Italiano,
Algoritmi e strutture dati 2/ed, McGraw-Hill



Testi consigliati



- Alan Bertossi, Alberto Montresor, *Algoritmi e strutture di dati 3/ed*, Città Studi Edizioni
 - I contenuti sono pressoché equivalenti a quelli del testo adottato (molti esercizi, con tanto di soluzioni)



- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduzione agli algoritmi e strutture dati 3/ed*, McGraw-Hill
 - Per chi vuole approfondire

Competenze richieste

- Per seguire con successo l'insegnamento è necessario aver acquisito le conoscenze e competenze relative al corso di **Programmazione**
 - In altri termini, in questo corso non imparate a programmare, ma dovete saper già programmare (e saper usare un qualche linguaggio di programmazione imperativo)
 - Per mettere in esecuzione gli algoritmi che studieremo useremo Java, ma gli elementi di programmazione object-oriented che ci servono li ripasseremo (o ve li insegneremo)
- Servono inoltre nozioni di base di tipo **Matematico**:
 - Sommatorie, polinomi, disequazioni, semplici elementi relativi all'analisi di funzioni

Programma del corso

- Complessità asintotica degli algoritmi
- Strutture dati elementari (Liste, Pile, Code, Alberi...)
- Alberi di ricerca
- Tabelle Hash
- Algoritmi di ordinamento e ricerca
- Tecniche Algoritmiche (divide et impera, algoritmi greedy, programmazione dinamica)
- Algoritmi su grafi (spanning tree, cammini minimi)
- Teoria dell'NP-completezza

Riguardo all'utilizzo di Java

- Useremo il linguaggio di programmazione Java
 - per fare esperimenti sugli algoritmi e le tecniche algoritmiche che studieremo
- Non utilizzeremo tecniche di programmazione avanzate, ci limiteremo alla:
 - Programmazione imperativa (che conoscete già)
 - Utilizzo delle Classi (che ripasseremo)
 - Uso delle Interfacce e dei Generici (che vi insegneremo)

Modalità d'esame

- Esame scritto: (2/3 del voto finale)
 - Sei appelli annuali (tre in sessione estiva, una in sessione autunnale, due in sessione invernale).
 - Scritto con esercizi
 - Vedremo esempi di esercizi e relative soluzioni durante le esercitazioni
 - Il libro di testo ed i libri consigliati contengono molti esempi di esercizi (in molti casi vengono presentate anche le soluzioni)
- Progetto: (1/3 del voto finale)
 - Lavoro da farsi in gruppo (due membri)
 - Verranno fissati su AlmaEsami specifici appelli per la discussione del progetto (con consegna da farsi prima della discussione)
- Ogni anno i docenti potrebbero cambiare
 - Siete invitati a sostenere le due prove entro Febbraio 2023

Progetto (M,N,K)-game

- (M,N,K)-game è una versione generalizzata del classico *tris*:
 - La partita viene giocata su una matrice di dimensione $M \times N$
 - Per vincere bisogna allineare K simboli consecutivi
 - Il tris è un (3,3,3)-game
- Il goal del progetto è sviluppare (in Java) un giocatore software per giocare al (M,N,K)-game
- Vi verrà fornito tutto il gioco (incluse le implementazioni di un paio di giocatori software banali) e le specifiche per poter implementare la vostra versione ottimizzata/ottimale
- Maggiori dettagli a metà corso, circa