

CORSO DI **ALGORITMI E STRUTTURE DATI**

Prof. ROBERTO PIETRANTUONO

Syllabus - A.A. 2022/23

Orario.....	1
Obiettivi del corso	1
Prerequisiti.....	1
Argomenti trattati	1
Materiale.....	2
Homeworks, esercitazioni, prove in itinere, esame.....	2
Homeworks	2
Esercitazioni e prove in itinere	3
Esame	3

Orario

Lunedì, 14.30-16.30
Martedì, 12.30-14.30
Giovedì, 12.30-14.30

Obiettivi del corso

Il corso intende fornire gli strumenti necessari per la progettazione e l'analisi di algoritmi e strutture dati. Al termine del corso, gli studenti:

- saranno in grado di analizzare la correttezza e le prestazioni asintotiche degli algoritmi in termini di complessità temporale;
- acquisiranno familiarità con una ampia varietà di strutture dati ed algoritmi che risolvono problemi di carattere fondamentale nello sviluppo delle applicazioni informatiche;
- saranno in grado di applicare e combinare le principali tecniche e paradigmi di progettazione, nonché strutture dati avanzate, per la sintesi di algoritmi corretti ed efficienti.

Prerequisiti

Programmazione

Argomenti trattati

Concetti introduttivi: algoritmi e strutture dati, ricorsione, analisi e progettazione degli algoritmi.

PRIMA PARTE

Tecniche di analisi e strutture dati elementari.

Analisi di correttezza: invariante di ciclo, correttezza di algoritmi ricorsivi.

Analisi di complessità: analisi asintotica, notazioni O , Ω , Θ ; analisi di algoritmi ricorsivi.

Strutture dati elementari: dizionari; pile e code; code di priorità; liste; tabelle hash e stringhe; alberi binari di ricerca.

Tecniche di progettazione. Classificazione di problemi, caratteristiche della soluzione, tecniche di progettazione: [divide et impera](#); [ricerca combinatoriale](#); [metodi euristici](#); [programmazione dinamica](#).

DIVIDE et IMPERA

Problemi ed algoritmi comuni.

Ordinamento: *mergesort*, *heapsort*, *quicksort*, ordinamento in tempo lineare (*counting sort*, *radix sort*, *bucket sort*), mediane e statistiche d'ordine.

RICERCA COMBINATORIALE E METODI EURISTICI

Ricerca esaustiva, ricerca combinatoriale; *backtracking*, *pruning* della ricerca.

Metodi euristici, ricerca locale ed algoritmi golosi ("*greedy*"). Cenni ad algoritmi meta-euristici, strategie di *evolutionary computation* (cenni)

Problemi ed algoritmi comuni.

Problemi combinatoriali (costruzione di subset e permutazioni), copertura minima di un insieme. Codici di Huffman. Il problema della selezione di attività. Il problema dello *job scheduling*.

PROGRAMMAZIONE DINAMICA

Introduzione alla programmazione dinamica. Ricerca esaustiva vs. ricerca *greedy* vs. programmazione dinamica. Applicazioni.

Problemi ed algoritmi comuni.

Problemi di *pattern matching* e *string matching*, il problema della *longest increasing sequence*, *edit distance*. Fibonacci. Problemi di partizionamento. Problema dello zaino. Ulteriori esempi.

SECONDA PARTE

Tecniche di analisi e strutture dati avanzate.

Strutture dati e tecniche di analisi avanzate: alberi RB, B-Alberi, alberi auto-aggiustanti, *heap* di fibonacci, strutture dati per insiemi disgiunti. Grafi. Rappresentazione, esplorazione in ampiezza e profondità, ordinamento topologico. Applicazioni. Tecniche di analisi di algoritmi avanzate: analisi ammortizzata.

COMBINAZIONE DI TECNICHE

per la risoluzione di categorie comuni di problemi:

Problemi ed algoritmi comuni.

1. Problemi su stringhe ed insiemi: (*approximate*) *string matching*, codifica e compressione, crittografia, copertura di insiemi (*set cover* e *set packing*), minimizzazione di automi a stati finiti.
2. Problemi di teoria dei numeri (es.: algoritmi DES ed RSA), problemi aritmetici ed algebrici.
3. Problemi combinatoriali: generazione di permutazioni, di sottoinsiemi, di partizioni.
4. Problemi su grafi: alberi di connessione minimi, ricerca di cammini minimi.

Cenni ad ulteriori tecniche per la progettazione.

- Multithreading e parallelismo: progettazione di algoritmi multithread, algoritmi paralleli e distribuiti. Esempi (Fibonacci, ordinamento).

Problemi intrattabili. Introduzione a problemi NP ed NP-completi. Soddisfacibilità, riducibilità. Esempi di problemi NP-completi.

Traduttori ed interpreti: analisi lessicale, analisi sintattica, analisi semantica, interpreti, strutture dati usate nei traduttori.

Parte Esercitativa: Prevalentemente in C.

Materiale

LIBRO DI TESTO ADOTTATO

1) Cormen, Thomas, Charles Leiserson, Ronald Rivest, and Clifford Stein. Introduction to Algorithms. 3rd ed. MIT Press, 2009. ISBN: 9780262033848.

TRASPARENZE DALLE LEZIONI ED ESERCITAZIONI

Lezioni ed esercitazioni del corso disponibili sul sito web docenti di Ateneo:

<http://www.docenti.unina.it/roberto.pietrantuono> (area download).

LIBRI CONSIGLIATI

2) S. Skiena. The Algorithm Design Manual, 3rd ed, Springer, 2020. ISBN-13: 978-3030542559, ISBN-10: 3030542556.

Homeworks, esercitazioni, prove in itinere, esame

Homeworks

Saranno assegnati degli *homework*, che devono essere consegnati durante il corso. Gli *homework* sono 3 set di esercizi/problemi, **HW #1-HW #3**. Gli homework contribuiscono alla valutazione per un totale di **9 punti**.

Le consegne devono essere rispettate; non saranno accettate consegne in ritardo. Gli homework possono essere svolti (e si consiglia di farlo) in coppia. I gruppi non possono essere composti da più di 2 persone.

Esercitazioni e prove in itinere

Le esercitazioni e le prove in itinere saranno delle “*programming challenges*”, secondo lo schema tipo adottato nei “*contest*” internazionali organizzati da università, associazioni, ed aziende di riferimento del settore IT. Le *challenges* saranno svolte in aula.

Per la valutazione verranno utilizzati arbitri online, comunemente usati nelle suddette *challenges* e *contest*, che saranno presentati a lezione, per avere una valutazione oggettiva (gli algoritmi implementati dovranno superare i casi di test).

Le *challenges* svolte come esercitazione in aula saranno 3. A queste si aggiunge una ulteriore challenge che varrà come prova in itinere. Infine, l'ultima challenge sarà la prova finale, svolta in sede d'esame. La prova in itinere e la prova finale contribuiscono alla valutazione finale per **9 punti ciascuna**.

Esame

La valutazione finale è determinata dai suddetti homeworks e prove secondo quanto riportato sopra, e sintetizzato nel seguente schema:

Attività	Punteggio massimo
Homework	9 (3 per ogni set di HW)
Prova in itinere	9
Prova finale, in sede d'esame	9
Prova orale	4

E' importante rimarcare che:

- Le valutazioni ottenute per gli *homework* e per le prove scritte possono cambiare radicalmente in sede di orale, se ad esempio si riscontra che esse non sono state svolte in autonomia o che comunque non si è in grado di spiegare in maniera adeguata quanto svolto.
- La prova orale e la prova finale sono in ogni caso obbligatorie (se anche si dovesse raggiungere un punteggio soddisfacente per lo studente con la sola prova in itinere e/o *homework*, la prova finale la prova orale è comunque obbligatoria). Di conseguenza, il requisito minimo per accedere all'orale è ottenere 14 punti nelle prove precedenti.

Policy per homework non consegnati e/o prova in itinere non svolta.

- All'atto della prenotazione (da 21 a 14 giorni prima dell'esame), lo studente può contattare il docente per ricevere un set di *homework* da svolgere (può essere richiesta anche solo una parte degli *homework*, se ad esempio sono state saltate delle consegne durante il corso); tali *homework* andranno consegnati entro 7 giorni prima dalla data d'esame. In caso di *homework* non consegnati, lo studente può accedere comunque all'esame.
- Il risultato della prova in itinere e degli homework svolti durante il corso sarà tenuto fino a Marzo.
- La prova in itinere può essere recuperata/ripetuta in sede d'esame, nella finestra di una settimana prevista per l'appello. Il risultato della prova in itinere ripetuta in un dato appello sostituisce quello della prova in itinere eventualmente svolta durante il corso (sia esso peggiore o migliore). La prova finale (più orale) conclude l'esame e presuppone l'acquisizione di tutto il programma; questo vuol dire che, se si non si supera l'esame, alla successiva seduta non saranno considerati i risultati parziali della seduta precedente (ossia, eventuale recupero della prova in itinere, prova finale ed homework); continueranno invece, fino a Marzo, ad essere validi i risultati della prova in itinere ed homework svolte durante il corso, se lo si desidera.
- Per prenotare il recupero della prova in itinere, si dovrà compilare il foglio Excel che sarà reso disponibile nel Team, oltre che prenotarsi anche tramite Segrepass per sostenere l'esame;