

Algoritmi e Strutture Dati

Lezione 1

28 settembre 2022

Informatica \equiv ~~Computer~~ Science
COMPUTING

Algoritmi

Strategie o procedimenti per la risoluzione di problemi

- Molto spesso dobbiamo risolvere problemi
- Se lo stesso problema si ripresenta e abbiamo già trovato una strategia per risolverlo, la possiamo applicare di nuovo
- Se non abbiamo già affrontato personalmente un certo problema, forse c'è qualcuno che l'ha già fatto e conosce una strategia per risolverlo
- Ci possono essere diverse strategie per risolvere lo stesso problema: meglio trovare una strategia "buona"
- Ci sono alcuni problemi che non si possono risolvere, altri sono molto difficili da risolvere

CLASSIFICAZIONE

Algoritmica

Parte dell'informatica che si occupa di tutti gli aspetti legati agli algoritmi

- *Progettazione* degli algoritmi
- Studio delle *strutture dati* da essi utilizzate
- *Analisi* della loro efficienza
- Studio delle *limitazioni inerenti e complessità dei problemi*
- Definizione di *nuovi modelli di calcolo*

Progettazione / correttezza
/ efficienza

ANALISI
DEGLI ALGORITMI
complessità

→ RISORSE

/ Tempo
/ Spazio
/ Mem
/ Processori
/ Energia

STRUTTURE DATI /

METODOLOGIE DI PROGETTAZIONE

COMPLETO VIAGGIATORE

n città

$$n \cdot (n-1) \cdot (n-2) \dots \cdot 1 = n!$$

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$$

$$5! = 120$$

$$8! = 40320$$

$$10! = 3\,628\,800$$

Algoritmi di
approssimazione

Perché studiare l'algoritmica?

ASPETTI PRATICI

Risolvere problemi

" " in modo efficiente

imparare a progettare

analizzare l'efficienza

computer inutili senza algoritmi

ASPETTI TEORICI / METODOLOGICI

Base / centro dell'informatica

Strumento mentale
per capire problemi
in varie aree

Obiettivi dell'insegnamento

- Impossibile studiare tutti i possibili algoritmi e immaginare tutti i possibili problemi
- Presentazione di alcuni algoritmi e strutture dati significativi
- Studio delle tecniche fondamentali di progetto e di analisi
- Studio di alcune problematiche tipiche
(in particolare relativamente ai problemi difficili)

Programma di massima

Lezioni

■ Introduzione e motivazioni

Algoritmi e problemi. Progetto e analisi di algoritmi.

Modello di calcolo RAM: criteri di costo uniforme e logaritmico.

Notazioni asintotiche.

■ Strutture indicizzate

Ricerca sequenziale e ricerca binaria.

Algoritmi elementari di ordinamento.

■ Ricorsione e tecnica divide-et-impera

Tecniche di ordinamento basate su ricorsione: mergesort, quicksort

Prodotto di matrici (algoritmo di Strassen)

Analisi di algoritmi ricorsivi.

■ Strutture collegate

Liste, Pile, Code, Alberi

Visite ad alberi

Alberi di ricerca (binari, AVL, 2-3, B-alberi)

Ordinamento basato su alberi: Heapsort

Gestione di partizioni: Tecniche Union-find

...

Programma di massima

...

- Tecniche hash
- Problemi di ottimizzazione e tecnica greedy
- Programmazione dinamica
- Grafi
Rappresentazione, Visite, Raggiungibilità, Cammini minimi, ...
- Problemi "difficili"
Introduzione alla NP-completezza

Laboratorio

- Implementazione di algoritmi e strutture dati

Materiale di riferimento

- Sito web: <https://gpighizziniasd.ariel.ctu.unimi.it/>
Avvisi, Diario delle lezioni, Materiale integrativo, Esercizi,
Registrazioni delle lezioni (a.a. 2020/21), ...
- Libro di testo:
 - C. Demetrescu, I. Finocchi, G. Italiano,
Algoritmi e strutture dati, McGraw-Hill, 2008.
- Per alcuni argomenti:
 - A. Bertoni, M. Goldwurm, *Progetto e analisi di algoritmi*,
Rapporto interno n. 230-98, Dip. Scienze dell'Informazione,
Università degli Studi di Milano.
 - Materiale preparato dal docente e pubblicato sul sito web

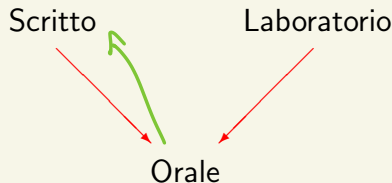
...

Materiale di riferimento

■ Altri riferimenti bibliografici:

- A.V. Aho, J.E. Hopcroft, J.D. Ullman,
The design and analysis of computer algorithms,
Addison-Wesley, 1974
- T. Cormen, C. Leiserson, R. Rivest, C. Stein,
Introduzione agli algoritmi e strutture dati, McGraw-Hill, 2010
- D.E. Knuth,
The Art of Computer Programming, 1968 – ...

Modalità d'esame



- Appelli: Gennaio, Febbraio (inizio), Febbraio (fine), Giugno, Luglio, Settembre
- Le prove sono valide per l'anno accademico in corso (cioè fino all'appello di settembre)
- Propedeuticità:
 - E' *necessario* avere superato l'esame di Programmazione
 - E' *opportuno* avere superato gli esami di Matematica del continuo, Matematica del discreto, Architettura