

ALGORITMI e strutture dati. ^{libero Longo}
21 Feb 2022
GianLuigi Zavattaro Modulo 1
Pietro di Lena Modulo 2
13 settimane di lezione
1° anno programmazione e architetture
Linguaggio

AFFRONTARE IL PROBLEMA CAPIRE COME RISOLVERLO
capire qual'è la risoluzione dei problemi migliore.

gianluigi.zavattaro@unibo.it
pietro.dilena@unibo.it

Non un ora fissa per ricevimento, su appuntamento.

strumenti VIRTUALE (slide ed esercizi)
e prove d'esame

- Camil Demetrescu per "ordine algoritmi"
- Alan Bertossi per esercizi con soluzioni.

Java. PROGRAMMA DEL CORSO

Modulo 2 (fatto prima) con Pietro di Lena

- Complessità asintotica degli algoritmi
- Strutture dati elementari (Liste, Pile, Code, Alberi...)
- Alberi di ricerca
- Tabelle Hash
- Algoritmi di ordinamento e ricerca

Modulo 1 (fatto dopo) con Giuligi Zavattoni

- Tecniche Algoritmiche
 - ← divide et impera
 - ← algoritmi greedy
 - ← programmazione dinamica
- Algoritmi su grafi
 - ← spanning tree → albero di copertura
 - ← cammini minimi
- Teoria dell' NP-completudine.

ESAME

Esame Scritto (2/3 voto Finale)

Progetto (1/3 voto finale) (2 membri)

Ogni anno i docenti potrebbero cambiare

- Siete invitati a sostenere le due prove entro Febbraio 2023.

non è vietato fare il progetto da soli.

al progetto segue la discussione media pesata delle due parti.

fissati un appello virtuale per la verbalizzazione.

slide 12. per il progetto
metà Marzo

slide e descrizione.

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
- 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.

cercate di usare l'ultima versione di Java.
10 min di time-out.
possiamo usare qualsiasi IDE, riga di comando.

obiettivo

stimare la mossa migliore per vincere, o non perdere la partita

algoritmi

PROCEDURA PER RISOLVERE UN DATO PROBLEMA
IN UN NUMERO FINITO DI PASSI.

origine della parola

matematico persiano (780-850 d.C.)

Muhammad ibn Musa al-Khwarizmi
latinoizzato in
algorithmi.

un algoritmo è una descrizione ad alto livello, un programma è una implementazione.

gli algoritmi esistono da prima dell'avvento dei computer, un programma è legato a un linguaggio, un algoritmo è scritto in pseudocodice.

Un algoritmo non può essere eseguito su un computer, un programma può essere eseguito su un computer. Per un algoritmo possiamo assumere un quantitativo illimitato di memoria, per un programma dovremmo fare check sulla memoria poiché limitata.

PROSPETTIVA STORICA

Euclide, Ezatodene, al-Kwarizmi:

PER DEFINIRE UN ALGORITMO:

input

output

sequenza di istruzioni

MA NOI PARTIAMO DAL PROBLEMA

input e output definito dal problema.

sequenza di istruzioni è variabile.

esempio algoritmo di Euclide.

Massimo Comun Divisore (300 a.C.)

ALGORITMO DI EUCLIDE

APPROCCIO UMANO

1. Dividi x per y e chiama r il resto della divisione
2. Se $r=0$ allora y è il MCD tra x e y
3. Se $r \neq 0$ allora assegna y ad x , r ad y e ritorna al punto 1.

IO Paolo Di Lena

Function $MCD(INT\ x, INT\ y) \rightarrow INT$

if $y == 0$ then
return x

else

$r = x \bmod y$
return $MCD(y, r)$

APPROCCIO

PSEUDOCODICE

↑ la sintassi potremmo inventarla, chiaro gli step.

decido di specificare il tipo di input e output altrimenti non si capisce

facciamo in modo che lo pseudocodice sia leggero (senza molti dettagli per la sintassi che lo appesantiscono) ma che renda chiaro la sequenza di istruzioni.

