

Introduzione alla computabilità

Stefano Mizzaro

Dipartimento di matematica e Informatica
Università di Udine
http://www.dimi.uniud.it/~mizzaro
mizzaro@dimi.uniud.it
26 maggio 2003

www.tecnologieweb.com

Stefano Mizzaro - Computabilità 2/30

Scaletta

- Problemi, domande, risposte
- Algoritmi, programmi, linguaggi di programmazione
- Algoritmi che calcolano funzioni
- Funzioni computabili e non

Stefano Mizzaro - Computabilità 3/30

Cosa fa un programma?

- Fornisce **risposte** a **domande**
 - Qual è la radice quadrata di 2345?
 - Qual è il massimo comun divisore di 234 e 36?
- Fornisce risposte a una **classe di domande**
 - Programma per calcolare la radice quadrata di un numero
- “Risolve **problemi**”

Stefano Mizzaro - Computabilità 4/30

Definizioni (1/2)

- **Problema** = insieme di domande omogenee
 - Esempi:
 - P1 = Calcolare $\sqrt[n]{n}$
 - P2 = Calcolare $mcd(x,y)$
- **Domanda** (“istanza di un problema”)
 - Esempi:
 - I1 = Quanto vale $\sqrt{341}$?
 - I2 = Quanto vale $mcd(48,36)$?

Stefano Mizzaro - Computabilità 5/30

Problemi e domande

Stefano Mizzaro - Computabilità 6/30

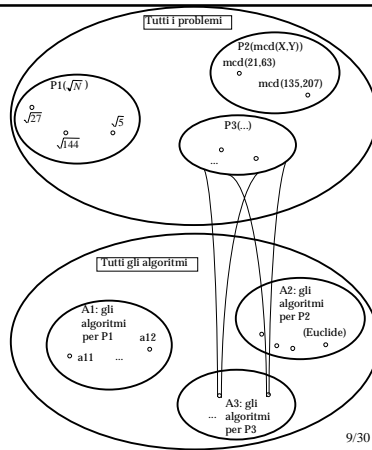
Definizioni (2/2)

- **Risposta** (a una domanda) = “soluzione di un’istanza”
- **Soluzione di un problema** = metodo generale che permette di fornire in modo uniforme la risposta a tutte le istanze di un problema. **Algoritmo risolvete**

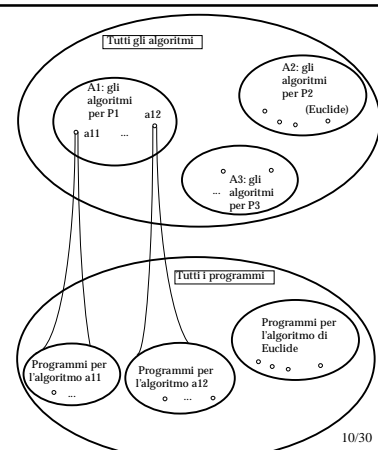
Algoritmi e programmi

- **Algoritmo**
 - “Metodo”, “procedimento”, “sequenza di passi”
 - Astratto
 - Non si può toccare/vedere
- **Programma**
 - Rappresentazione precisa di un algoritmo in un certo **linguaggio di programmazione**
 - Per eseguire/comunicare un algoritmo bisogna rappresentarlo!!

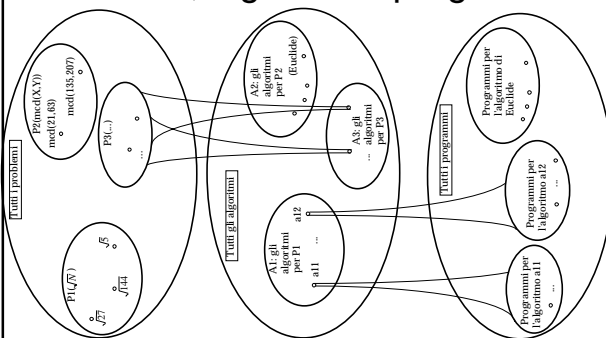
Problemi e algoritmi



Algoritmi e programmi



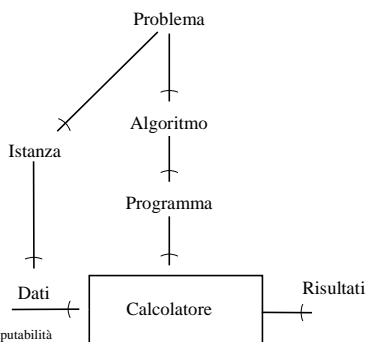
Problemi, algoritmi e programmi



Il calcolatore

- È un:
 - Esecutore di algoritmi
 - rappresentati da programmi
 - mediante un linguaggio di programmazione
- Esecutore **universale** di algoritmi
- Un algoritmo va eseguito su dati
 - Es.: calcola $\text{sqrt}(897)$

Calcolatore, programma e dati



Stefano Mizzaro - Computabilità

13/30

Finitezza, univocità, effettività

- Un programma/algorithmo deve essere:
 - Finito (# finito di istruzioni)
 - Univoco, non ambiguo ($x = x + \text{"un po' "}$)
 - Effettivo: deve esistere un calcolatore in grado di eseguire le istruzioni

Stefano Mizzaro - Computabilità

14/30

Curiosità legittime...

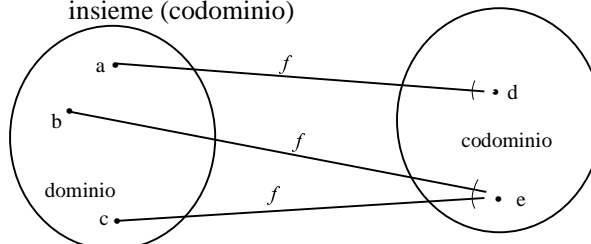
- Ma con gli algoritmi si può fare tutto?
- E se no, che cos'è che si può fare?
- E quello che si può fare dipende dal linguaggio di programmazione scelto?
- A queste domande risponde la **teoria della computabilità (calcolabilità)**
- Approccio formale, matematico, preciso

Stefano Mizzaro - Computabilità

15/30

Funzioni

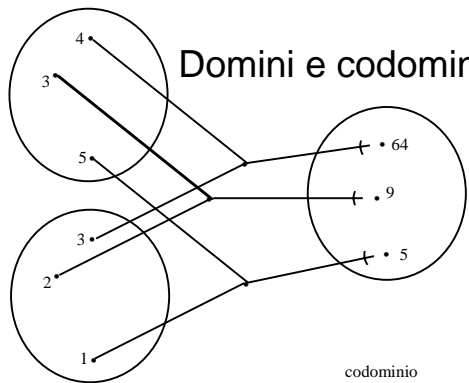
- Funzione = legge che associa a un elemento di un insieme (dominio) un elemento di un insieme (codominio)



Stefano Mizzaro - Computabilità

16/30

Domini e codomini



Stefano Mizzaro - Computabilità

17/30

Algoritmi calcolano funzioni

- Algoritmo per calcolare la radice quadrata calcola la funzione che associa a un numero la sua radice quadrata
- Algoritmo di Euclide calcola la funzione che dati 2 numeri restituisce il loro MCD
- ...

Stefano Mizzaro - Computabilità

18/30

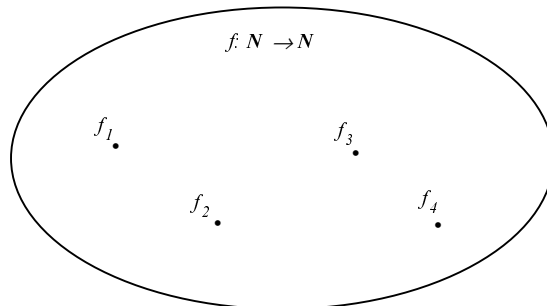
Quali funzioni?

- Funzioni $f: R \rightarrow R$: analisi matematica
- La teoria della computabilità si occupa delle funzioni $f: N \rightarrow N$
- Parziali
 - Ad esempio programma con un ciclo infinito che non dà nessun output
 - Funzione "non definita"

Stefano Mizzaro - Computabilità

19/30

Le funzioni $f: N \rightarrow N$



Stefano Mizzaro - Computabilità

20/30

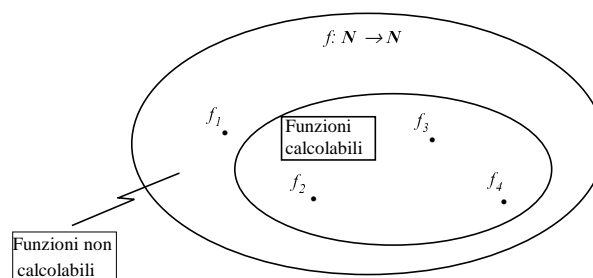
Curiosità legittime...

- Def.: **Funzione calcolabile (computabile)** = funzione per cui esiste un algoritmo che la calcola
- Ora, finalmente, possiamo porci una domanda precisa:
Le funzioni $f: N \rightarrow N$ sono tutte calcolabili?
- Data una qualsiasi funzione, esiste sempre un algoritmo che la calcola?

Stefano Mizzaro - Computabilità

21/30

No! Funzioni calcolabili e non



Stefano Mizzaro - Computabilità

22/30

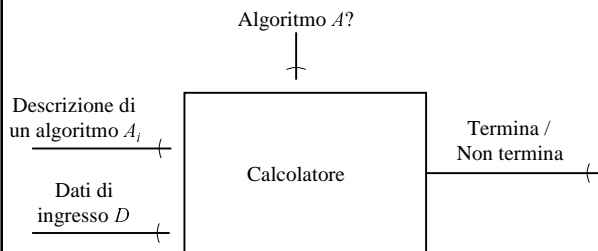
No? Voglio un esempio...

- Il problema della terminazione (*Halting problem*)
 - Trovare un algoritmo A in grado di dirci se un qualsiasi algoritmo A_i su dati D termina.
- (un algoritmo potrebbe non terminare perché entra in un ciclo infinito)

Stefano Mizzaro - Computabilità

23/30

Il problema della terminazione



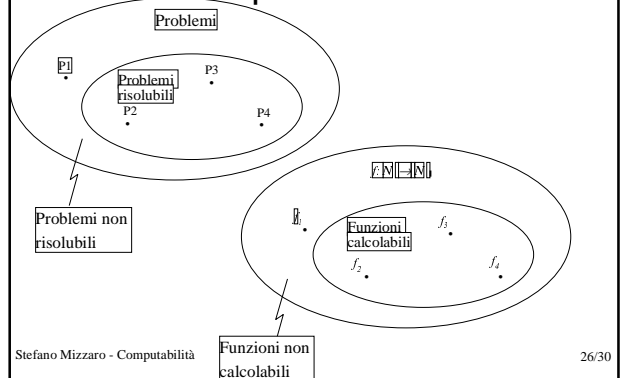
Stefano Mizzaro - Computabilità

24/30

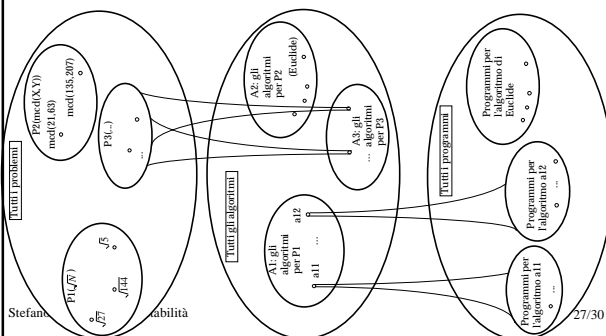
Una soluzione?

- Beh, potrei far eseguire A_i sui dati e attendere
- Se termina so che termina
- Se dopo 2 ore non ha terminato... so che non ha terminato in due ore... ma quanto devo aspettare?!
- Non va!
- Teorema: A non esiste

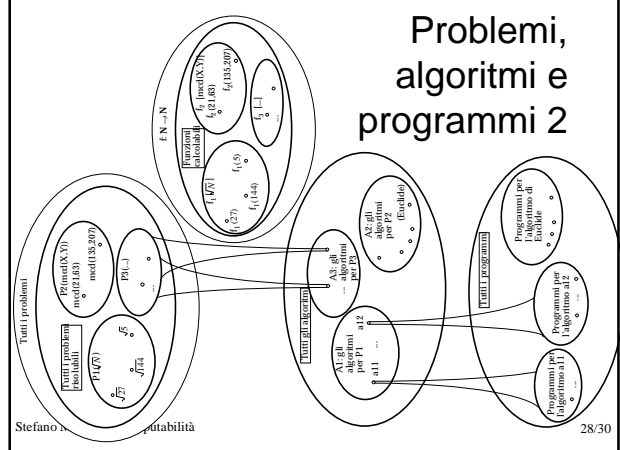
Funzioni non calcolabili e problemi non risolvibili



Problemi, algoritmi e programmi 1



Problemi, algoritmi e programmi 2



Riferimenti

- <http://www.dimi.uniud.it/~mizzaro/papers/algorithmo.pdf>
- Capp. 3 e 4
- Sostituire:
 - Ufficio
 - Impiegato
 - Diagramma di flusso

Riassunto

- Problemi, domande, risposte
- Algoritmi, programmi, linguaggi di programmazione
- Algoritmi che calcolano funzioni
- Funzioni computabili e non