



Progetto e analisi di algoritmi

Roberto Cordone

DTI - Università degli Studi di Milano

Polo Didattico e di Ricerca di Crema

Tel. 0373 / 898**089**

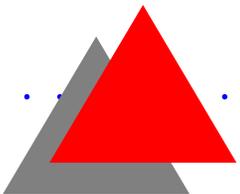
E-mail: **cordone@dti.unimi.it**

Ricevimento: **su appuntamento**

Web page: **<http://www.dti.unimi.it/~cordone>**

Lezioni: **Martedì dalle 11.00 alle 13.00**

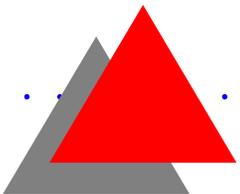
Giovedì dalle 11.00 alle 13.00





Progetto e analisi di algoritmi

- **Progetto** è passare dalla descrizione di un problema alla descrizione di un algoritmo che lo risolve
- **Analisi** è passare dalla descrizione di un problema e di un algoritmo
 1. alla prova che l'algoritmo risolve il problema o no (**correttezza**)
 2. alla prova di quanto costa risolvere il problema con l'algoritmo (**complessità**)
- **Algoritmo** è uno strumento formale (cioè meccanico) per risolvere un problema





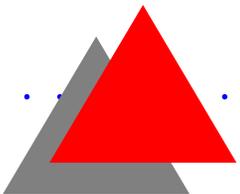
Principi del corso

1. $A \rightarrow P$

2. $\mathcal{A} \subset \mathcal{M}$

3. $A \neq C_A$

4. $P \rightarrow M \xrightarrow{A} Sol \rightarrow Str$



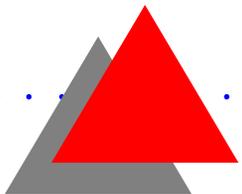


1. $A \rightarrow P$

Un algoritmo serve a risolvere un problema

... che cosa sono i problemi?

Pensiero	Azione
1. progetto per riduzione	1. progettare algoritmi senza pensare ad algoritmi
2. classi di complessità	2. progettare un solo algoritmo per problemi diversi
	3. sapere a priori che tipo di algoritmo cercare
	4. vincere un milione di dollari?



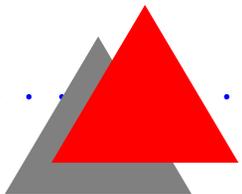


2. $A \rightarrow M$

Gli algoritmi sono macchine

- funzionano perché sono fatti in un certo modo
- sono fatti in un certo modo affinché funzionino

Pensiero	Azione
1. progetto per induzione	1. ricordare (concetto base)
2. invarianti di ciclo	2. descrivere in modo corretto e completo
3. analisi di complessità	3. implementare e riparare (debugging)
	4. analizzare (correttezza e complessità)
	5. progettare





3. $A \neq C_A$

Un algoritmo non è il codice che lo descrive

In passato, realizzare un algoritmo era

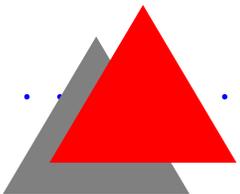
- collegare opportunamente ingranaggi, alberi a camme, ...
- collegare opportunamente cavi, relais, ...
- perforare opportunamente schede

Oggi, realizzare un algoritmo è

- scrivere codice in linguaggi di programmazione (es., C)
- collegare opportunamente registri e porte logiche

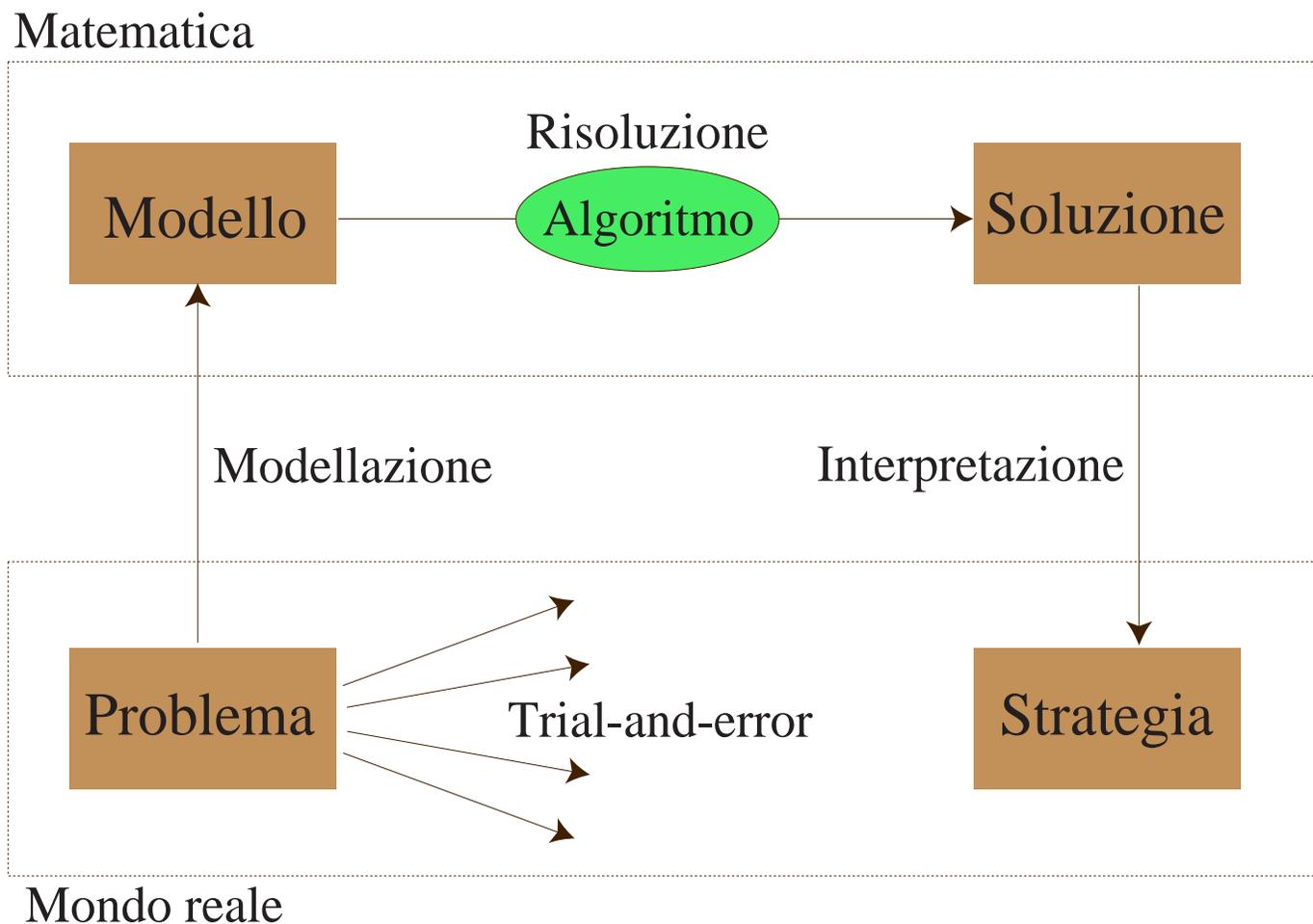
In futuro?

La tecnologia cambia, l'algoritmo rimane lo stesso

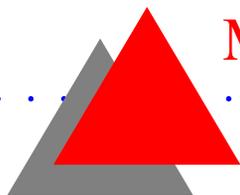




$$4. P \rightarrow M \xrightarrow{A} Sol \rightarrow Str$$



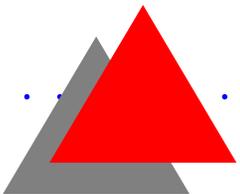
Modellazione e interpretazione sono le fasi più delicate e creative





Programma (1)

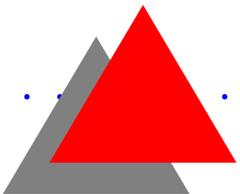
- **Elementi di teoria della computazione** (8 T + 1 E + 1 L)
 - Modelli computazionali
 - Complessità di algoritmi e problemi
 - Classi di complessità \mathcal{P} e \mathcal{NP}
 - Oracoli, trasformazioni, **riduzioni**
 - \mathcal{NP} -completezza
- **Progetto per induzione** (3 T + 1 E + 1 L)
 - **Principio di induzione** e numeri naturali
 - Esempi elementari di progetto per induzione
 - Un esempio ricorsivo complesso
 - **Equazioni ricorrenti**





Programma (2)

- **Invarianti di ciclo** (3 T + 1 E + 1 L)
 - Schema generale e algoritmi di ordinamento
 - Algoritmi per l'albero ricoprente minimo
 - Un algoritmo iterativo complesso
 - **Sommatorie approssimate**
- **Algoritmo greedy e matroidi** (1 T)
- **Implementazione di algoritmi** (4 L)
 - Liste
 - Grafi
 - Algoritmi di branching





Modalità d'esame

1. Fase **teorica** (16 punti):

esame scritto con domande a risposta aperta ed esercizi

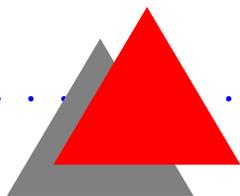
2. Fase **pratica** (14 punti) in gruppi da 1 o 2:

- progetto di un algoritmo e **stesura in C**
- stesura di una **relazione** in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
- realizzazione e commento di una **campagna prove**

Se tenete alla laurea, fate il progetto prima di cominciare la tesi!

Non c'è precedenza fra teoria e pratica e si può ripetere la teoria quante volte si vuole (vale l'**ultimo compito consegnato**)

Dopo un anno o si registra o si perde il lavoro





Consigli

1. **venire a lezione**
(i lucidi sono quasi completi, ma per natura superficiali)
2. **prendere appunti completi**, non glossare i lucidi
(se non sai scrivere quel che hai sentito, non hai capito)
3. **interrompere** per chiedere **chiarimenti** o **conferme** e segnalare **incomprensioni**
4. **studiare sugli appunti**
(il lucido ti fa credere di aver capito, ma stai solo ripetendo)
5. **studiare un po' per volta**, rivedendo più volte gli argomenti
(ogni volta scoprirai qualcosa di nuovo)

