

Tecniche della Programmazione, lez. 1

- **Qualche informazione sul corso, il Programma**
- Informatica, Calcolatore Elettronico, Algoritmo ...
- Introduzione allo sviluppo ed esecuzione di programmi

CORSO DI LAUREA IN INGEGNERIA DELL'INFORMAZIONE

dm270 (legge di riferimento)

FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE, INFORMATICA E STATISTICA

Gestione del Corso di Laurea

- Consiglio d'Area Didattica (CAD, CdS) ... i docenti ...
 - riconoscimento esami, trasferimenti, percorsi formativi, part-time, erasmus ... gestione orari e aule

sito web del Corso di Laurea (tramite ateneo)

- RAPPRESENTANTE DEGLI STUDENTI (vedi sito web Corso di Laurea)
- Percorso Formativo ... proposta presentabile dal 15/12/2025 an 31/3/2026

Segreteria Didattica (QUI ... traversata la piazza d'armi)

Bacheche (QUI ...)

Segreteria Studenti (traversata la strada ... il parcheggio ... il parco, ... a destra ...)

CORSO DI LAUREA IN INGEGNERIA DELL'INFORMAZIONE

dm270 (legge di riferimento)

FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE, INFORMATICA E STATISTICA

CLASSROOM del CORSO DI LAUREA [7pgrsjq](#) **iscrivetevi**

ELEZIONI RAPPRESENTANTI STUDENTI

- **3 rappresentanti**
- **2 candidati dal primo anno** sarebbero una buona cosa
- **ELEZIONI 20 Ottobre, via telematica**
(istruzioni via email dall'ateneo)

Programma del Corso TdP

Basi di Informatica, programmazione dei calcolatori elettronici

1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)

Funzionamento del Calcolatore,

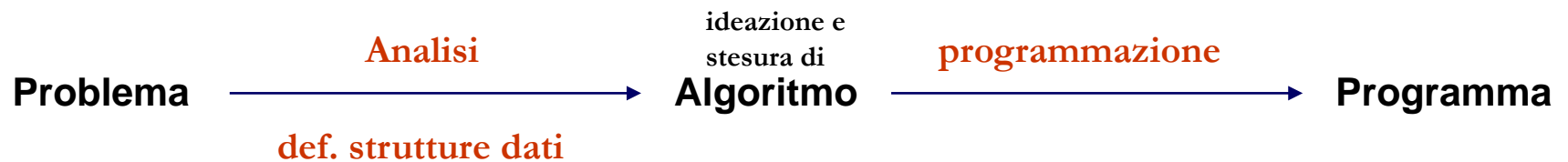
... da programma in **linguaggio ad alto livello** (... il C)

→ Compilazione → linking → "programma .exe" ...

... a programma in **linguaggio macchina**

→ Caricamento dell'eseguibile in memoria → **esecuzione**

2) Processo di sviluppo del software



3) Tecniche di programmazione e algoritmi interessanti

Programma del Corso (1/3)

1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)

Funzionamento del Calcolatore,

- da programma in **linguaggio ad alto livello** (... il C)

→ Compilazione → linking → ...

a programma in **linguaggio macchina**

→ caricamento dell'**eseguibile** in **memoria** →

→ esecuzione

... hardware o software?



Programma del Corso (1/3)

hardware con software

... hardware



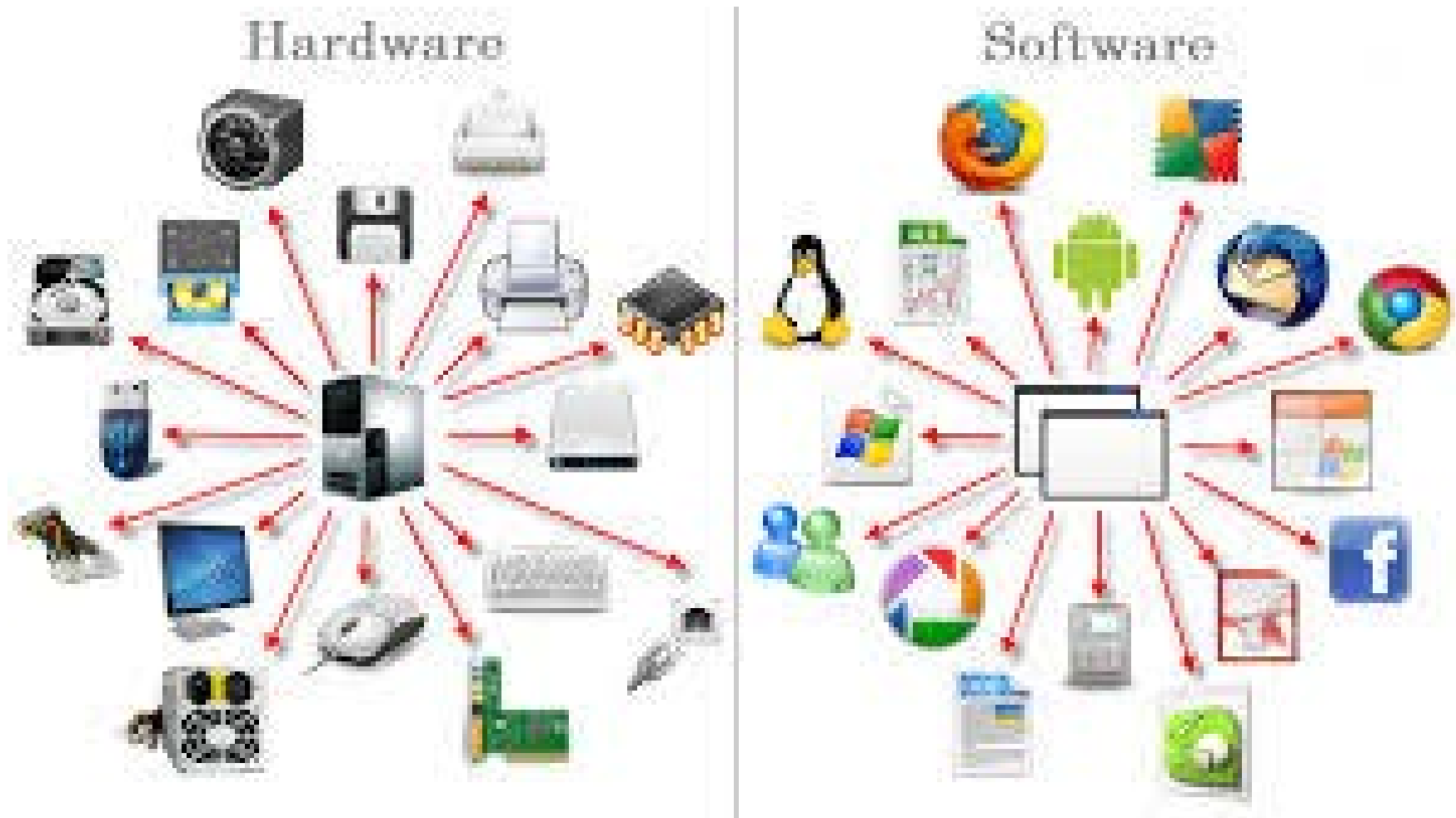
... hardware o software?



Programma del Corso (1/3)

Basi di Informatica, programmazione dei calcolatori elettronici

1)

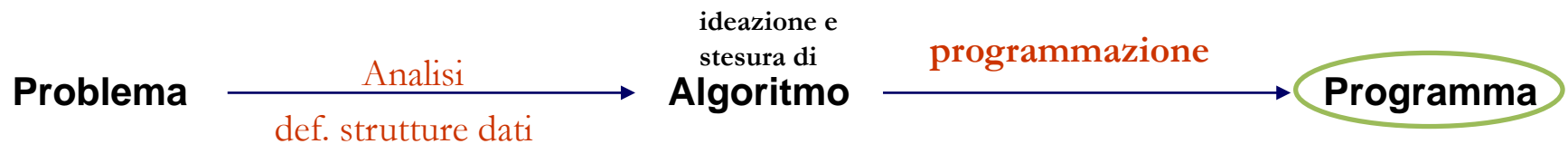


... hardware e software?

Programma del Corso (2/3)

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

2) Processo di sviluppo del software

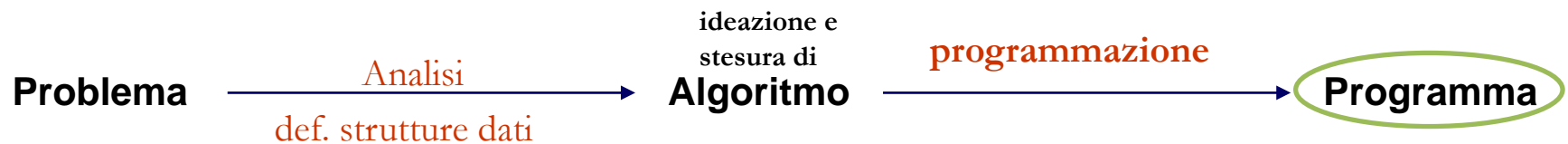


Scopo: gestire (far gestire dal calcolatore ...) le
informazioni che caratterizzano il problema
in modo da produrre
informazioni che caratterizzano la soluzione del problema

Programma del Corso (2/3)

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

2) Processo di sviluppo del software



Scopo: gestire (far gestire dal calcolatore ...) le **informazioni** che caratterizzano il problema in modo da produrre **informazioni** che caratterizzano la soluzione del problema

Scopo: gestire (far gestire dal calcolatore ...)

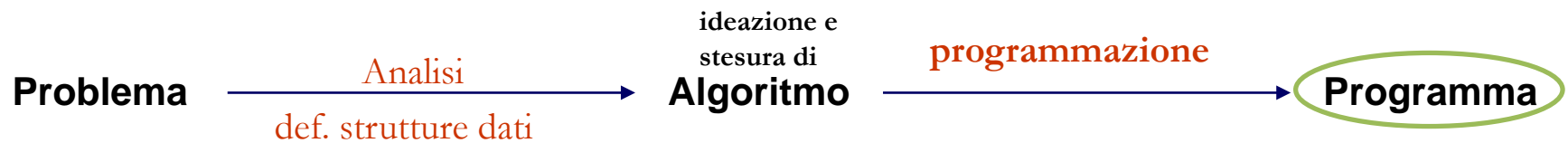
i DATI che rappresentano le informazioni che caratterizzano il problema in modo da produrre

i DATI che rappresentano le informazioni che caratterizzano la soluzione del problema

Programma del Corso (2/3)

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

2) Processo di sviluppo del software



Scopo: gestire (far gestire dal calcolatore ...)

i DATI che rappresentano le informazioni che caratterizzano il problema

in modo da produrre

i DATI che rappresentano le informazioni che caratterizzano la soluzione del problema

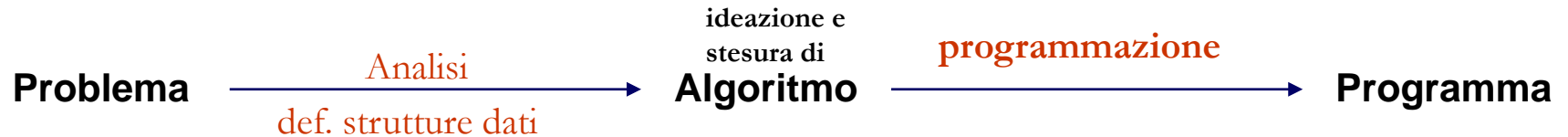
dato = quantità associata ad una **informazione**

struttura dati = modo formale di definire come sono rappresentate le **informazioni** dentro al programma (cioè dentro al calcolatore)

Programma del Corso (2/3)

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

2) Processo di sviluppo del software



dato = quantità associata ad una informazione

struttura dati = modo formale di definire come sono rappresentati i dati (cioè le informazioni) dentro al programma (cioè dentro al calcolatore)

algoritmo = serie di passi da eseguire, manipolando e producendo dati, con l'obiettivo di produrre i dati che rappresentano la soluzione del problema

Programma del Corso (2/3) – mettiamo in ordine

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
- da programma in linguaggio ad alto livello (... il C)
 → Compilazione → linking → ...
 a programma in linguaggio macchina
 → caricamento dell'eseguibile in memoria →
 → esecuzione

in quale ordine dovrebbero essere le scatole qui sotto?

2) Processo di sviluppo del software



Scopo: il costo del caffelatte in euro è rappresentato da X - un numero reale; i soldi che ho, in euro, sono un dato chiamato Y (un altro reale)

Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

analisi del problema

dato = quantità ... se X è minore-uguale Y , allora il risultato è Sì, senno' è NO;

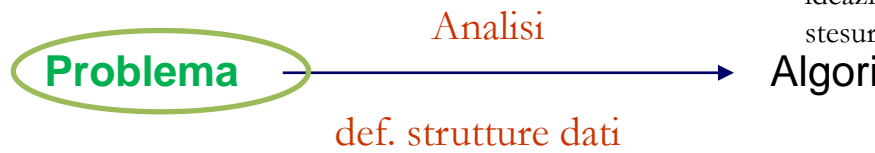
struttura dati = modo formale di definire come sono rappresentati i dati (cioè le informazioni) dentro al programma (cioè dentro al calcolatore)

algoritmo = serie di passi da eseguire, manipolando e producendo dati, con l'obiettivo di produrre i dati che rappresentano la soluzione del problema

Programma del Corso (2/3) – si descrive il problema

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

vari passi, tra cui ... se X è minore-uguale Y , allora il risultato è Sì, senno' è NO;



Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

il costo del caffelatte in euro è rappresentato da X - un numero reale; i soldi che ho, in euro, sono un dato chiamato Y - un numero reale;

informazioni che caratterizzano la soluzione del problema

dato = quantità associata ad una informazione

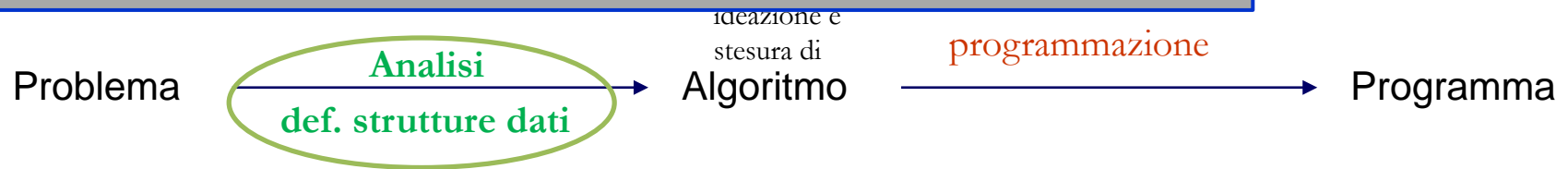
struttura dati = modo formale di definire come sono rappresentati i dati (cioè le informazioni) dentro al programma (cioè dentro al calcolatore)

algoritmo = serie di passi da eseguire, manipolando e producendo dati, con l'obiettivo di produrre i dati che rappresentano la soluzione del problema

Programma del Corso (2/3) – si analizza il problema

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
- da programma in linguaggio ad alto livello (... il C)
 → Compilazione → linking → ...
 a programma in linguaggio macchina
 → caricamento dell'eseguibile in memoria →
 → esecuzione

vari passi, tra cui ... se X è minore-uguale Y , allora il risultato è Sì, senno' è NO;



Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

il costo del caffelatte in euro è rappresentato da X - un numero reale;
i soldi che ho, in euro, sono un dato chiamato Y - un numero reale; per poter comprare il caffelatte ... bisogna fare due conti e confronti su questi dati. Non sembra serva altro, a giudicare dall'algoritmo che abbiamo in mente ...

Programma del Corso (2/3) – si sintetizza l' algoritmo

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
 - da programma in linguaggio ad alto livello (... il C)
 - Compilazione → linking → ...
 - a programma in linguaggio macchina
 - caricamento dell'eseguibile in memoria →
 - esecuzione

2) Processo di sviluppo del software



Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

il costo del caffelatte in euro è rappresentato da X
- un numero reale;
i soldi che ho, in euro, sono un dato chiamato Y - un numero reale; per poter comprare il caffelatte ...

Vari passi: quanto è X ? Quanto è Y ? ...
se X è minore-uguale Y , allora il risultato è Si ,
senno' è NO ;

Programma del Corso (2/3)

- 1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,
- da programma in linguaggio ad alto livello (... il C)
 → Compilazione → linking → ...
 a programma in linguaggio macchina

NB Se gli "identificatori" (i nomi) dei dati sono "evocativi" e` meglio ...

2) Processo di sviluppo del software



Scopo: gestire (far gestire dal calcolatore ...) le informazioni in modo da poter accedere alle informazioni

Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

del problema

dato = quantita` di
struttura dati = m
le informazi

il costo del caffelatte, in euro, e` rappresentato da **costoCaffelatte** - un numero reale;

l'informazione sui soldi che possiedo e` rappresentata da **soldiCheHo** - un numero reale;

oe`

algoritmo = serie
l'obiettivo c

vari passi, ... tra i quali ... se **costoCaffelatte** e` minore-uguale di **soldiCheHo**, allora il risultato e` Si` , senno` e` NO;

Programma del Corso (2/3)

1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)

il "metodo di calcolo" proposto si puo` applicare a qualsiasi istanza del problema, ottenendo il risultato corrispondente a quell'istanza

2) P

F istanza?

What's *istanza*? Dovrebbe stare sugli appunti ;)

h
ma

Problema: il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

il costo del caffelatte, in euro, e` rappresentato da **costoCaffelatte** - un numero reale;
l'informazione sui soldi che possiedo e` rappresentata da **soldiCheHo** - un numero reale;

vari passi, tra cuise **costoCaffelatte** e` minore-uguale di **soldiCheHo**, allora il risultato e` Si` , senno` e` NO;

Programma del Corso (3/3)

1) Architettura di base del Calcolatore (Memoria, CPU, IO, ...)
Funzionamento del Calcolatore,

- da programma in linguaggio ad alto livello (... il C)

→ Compilazione → linking → ...

a programma in linguaggio macchina

→ caricamento dell'eseguibile in memoria →

→ esecuzione

2) Processo di sviluppo del software

3) Tecniche di programmazione e algoritmi interessanti

3.1) "toolbox":

- strutture di controllo (**if, for, while** ...)

- strutture dati di base (**array, file** ...)

- **conoscenze algoritmiche** da riusare per **risolvere nuovi problemi** e per **apprendere altre** tecniche ed altri algoritmi

3.2) altre tecniche e METODOLOGIE

Programma del Corso (3/3)

3) Tecniche di programmazione e algoritmi interessanti

3.1) "toolbox":

- strutture di controllo (if, for, while ...)
- strutture dati di base (array, file ...)
- conoscenze da riusare per risolvere nuovi problemi e per apprendere altre tecniche ed altri algoritmi

3.2) altre Tecniche e Metodologie

- ricorsione (T)
- TEST (M)
- Ancora Sviluppo Programma (M)
- Gestione strutture dati piu` complesse (M)(T)

Strutture dati? Modo per rappresentare nella memoria del calcolatore gli oggetti del mondo reale coinvolti nel problema

Vedremo

Strutture dati STATICHE

e strutture dati DINAMICHE

PRIMA PARTE DEL CORSO: ... fino a 3.1 (compreso)

SECONDA PARTE: ... 3.2

Prima parte: nel primo semestre

LUN, Aula 15; MER. Lab  I;

Lab. a distanza? MER?

Tutore, alcuni MER

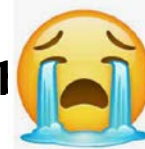
Google Classroom di TdP ... iscrivetevi!!!!

vedi pagina web con il
calendario delle lezioni

Seconda parte: nel secondo semestre

Bisogna seguire tutte e due le parti, di seguito ... vedi dopo

Lezioni + Esercitazioni "in laboratorio" (sul PC)



- Al termine del primo semestre
(Gen, Feb) **PROVA INTERMEDIA**
- Al termine del secondo semestre
(Giu, Lug e seguenti): **APPELLI DI ESAME**
(esame finale - scritto con orale - e verbalizzazione)

Organizzazione (2/2)

Prova intermedia = breve presentazione degli esercizi fatti durante il primo semestre e domanda (vedi pagina web)

- questa prova e` relativa ad una parte del programma
- ci sono date per prove intermedie anche successivamente a febbraio, **MA** e` meglio **seguire le lezioni e fare subito la prova intermedia a gennaio/febbraio.**

durante la seconda parte (e forse anche la prima)

- HOMEWORK (Compiti)

tre nel secondo semestre; da sottomettere via ...
vedi pagina web del corso TdP.

- sono occasioni per mettere alla prova cio` che si e` fatto fino ad un dato momento del corso; sono parte delle attivita` pratiche; la correzione puo` essere utile a capire se ci sono problemi ...

- servono anche per essere ammessi a fare l'esame finale ...

<http://www.diag.uniroma1.it/~marte>

seguire percorso per DIDATTICA - Tecniche della Programmazione

- **programma** (e sync sui testi)
- ELENCO LEZIONI (svolte/da svolgere)
- sistema per consegnare i compiti e vederne la valutazione
- Materiale didattico
- TESTI di esame (esempi)
- MESSAGGI/COMUNICAZIONI (ricevimento studenti)

TESTI di studio (**VEDI programma**)

- **Deitel&Deitel:**
Il linguaggio C, Fondamenti e tecniche di programmazione.
Ed. Pearson
- **Calvanese et al.** (da un certo punto nel secondo semestre)
Progetto di strutture dati in C.
Ed. Progetto Leonardo

Tecniche della Programmazione, lez. 1

- Qualche informazione sul corso, il Programma
- and now

Informatica, Calcolatore Elettronico, Algoritmo ...

- Introduzione allo sviluppo ed esecuzione di programmi

non e` "la scienza dei computer"

e` la scienza che studia

**come risolvere problemi mediante il calcolatore
cioe`**

- **come organizzare** le informazioni sul problema,
- **come rappresentarle** nel calcolatore,
- **come dare al calcolatore il "compito" di trattarle automaticamente**

Computer science is no more about computers than astronomy is about telescopes (E. W. Dijkstra)



e' la **scienza** 🥰 che studia
come risolvere problemi mediante il calcolatore
cioe` come organizzare le informazioni sul problema, come rappresentarle
nel calcolatore, come dare al calcolatore il "compito" di trattarle
automaticamente

- **descrizione, analisi del problema**
- **definizione degli strumenti logici che rappresentano, nel calcolatore, informazioni sul problema**
- **definizione delle sequenze di operazioni che deve fare il calcolatore per produrre una soluzione al problema**

scienza = approccio

rigoroso, disciplinato

sistematico

sincero

sperimentale

e' la **scienza** 🥰 che studia
come risolvere problemi mediante il calcolatore
cioe' come organizzare le informazioni sul problema, come rappresentarle
nel calcolatore, come dare al calcolatore il "compito" di trattarle
automaticamente

- **descrizione, analisi del problema**
- **definizione degli strumenti logici che rappresentano, nel calcolatore, informazioni sul problema**
- **definizione delle sequenze di operazioni che deve fare il calcolatore per produrre una soluzione al problema**

strutture dati

produzione di una soluzione e del programma che puo'
"applicare" la soluzione al problema
(algoritmo e programma)

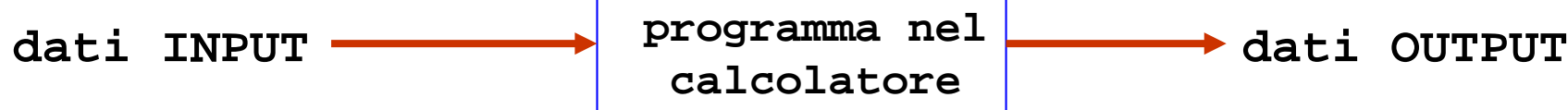
Informatica

- descrizione, analisi del problema
- definizione degli strumenti logici che rappresentano, nel calcolatore, informazioni sul problema
- definizione delle sequenze di operazioni che deve fare il calcolatore per produrre una soluzione al problema

produzione della soluzione teorica al problema



applicazione della soluzione ad una istanza del problema



Il Calcolatore Elettronico

Macchina che **memorizza, elabora, distribuisce i dati** che rappresentano le **informazioni** di un problema (e di un qualsiasi caso - istanza - in cui si presenta quel problema).



Per elaborare, memorizzare, distribuire ... esegue le **istruzioni** di un **programma**, che ha lo scopo di produrre il **risultato** (o i risultati) del problema

QUALI PROBLEMI ?

PROBLEMI DI INTERESSE SONO QUELLI PER I QUALI È POSSIBILE UNA
FORMALIZZAZIONE MATEMATICA

-- il caffelatte ha un costo; io ho dei soldi; posso comprare il caffelatte?

-- Disporre una sequenza di interi in ordine crescente

-- Trovare l'intero più grande in una sequenza di interi

-- Trovare il percorso più breve tra due nodi di una rete

-- date delle parole chiave, trovare le pagine web in cui se ne parla

-- data la mia posizione, trovare il percorso migliore per arrivare in segreteria studenti

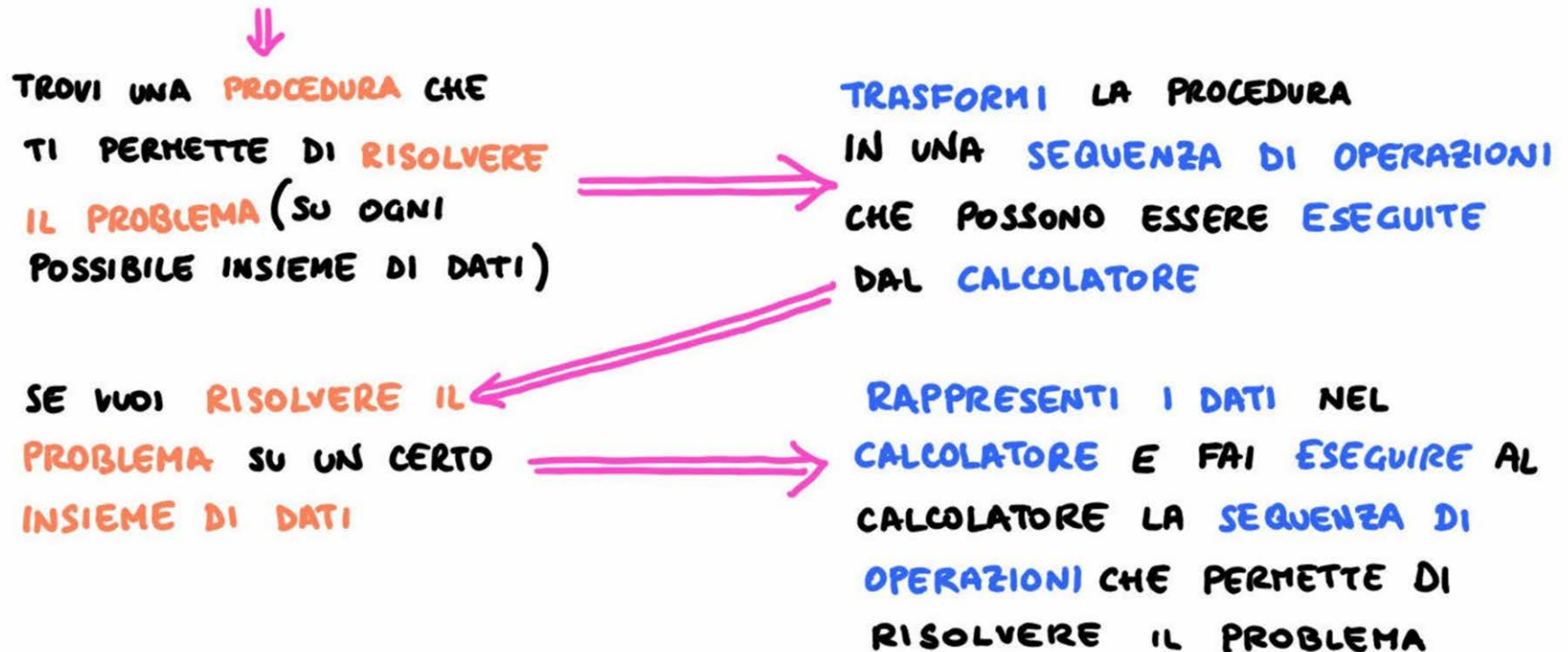
-- . . .

processo per arrivare a risolvere un problema con un programma

a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma

ESERCIZIO ... la soluzione e' nelle slide in fondo: non guardarle subito!



Istanza di un problema

IL caso particolare del problema, cui vorremmo applicare il nostro programma per ottenere il relative risultato

istanza1: ho 50 centesimi e il caffelatte costa un euro

istanza2: ho 2 euro e il caffelatte costa un euro

istanza3: ho 1 euro e il caffelatte costa un euro e 50



Istanza di un problema

IL caso particolare del problema, cui vorremmo applicare il nostro programma per ottenere il relative risultato

istanza1: ho 50 centesimi e il caffelatte costa un euro

istanza2: ho 2 euro e il caffelatte costa un euro

istanza3: ho 1 euro e il caffelatte costa un euro e 50



Istanza di un problema

IL caso particolare del problema, cui vorremmo applicare il nostro programma per ottenere il relative risultato

istanza1: ho 50 centesimi e il caffelatte costa un euro

istanza2: ho 2 euro e il caffelatte costa un euro

istanza3: ho 1 euro e il caffelatte costa un euro e 50



PROBLEMI, ALGORITMI, PROGRAMMI

ESERCIZIO ... la soluzione e la continuazione sono nelle slide in fondo: non guardarle subito!

PROBLEMI DI INTERESSE SONO QUELLI PER I QUALI È POSSIBILE UNA

[]

SI POSSONO RISOLVERE ATTRAVERSO UN [], CHE CONSISTE NELLA
sequenza di passi operativi [] CHE CONSENTE DI OTTENERE
I [] A PARTIRE DAI []

dopo

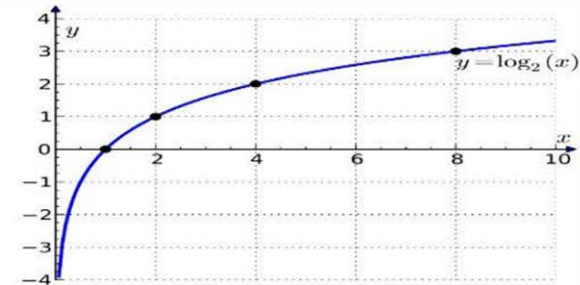
Algoritmo e Programma

ALGORITMO : COSA È ?

UN FILM
ALGORITHM-
THE HACKER
MOVIE



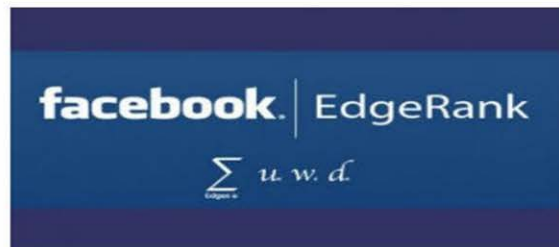
L'ANAGRAMMA DI LOGARITMO



UN DJ DI MUSICA
ELETTRONICA



DEVE ESSERE TRENDY
LO USA PURE ZUCKERBERG !



macche`

Al-Khwārizmī

Muḥammad ibn Mūsā al-Khwārizmī,
(circa 780-850),
Muslim mathematician and astronomer.

(Britannica)

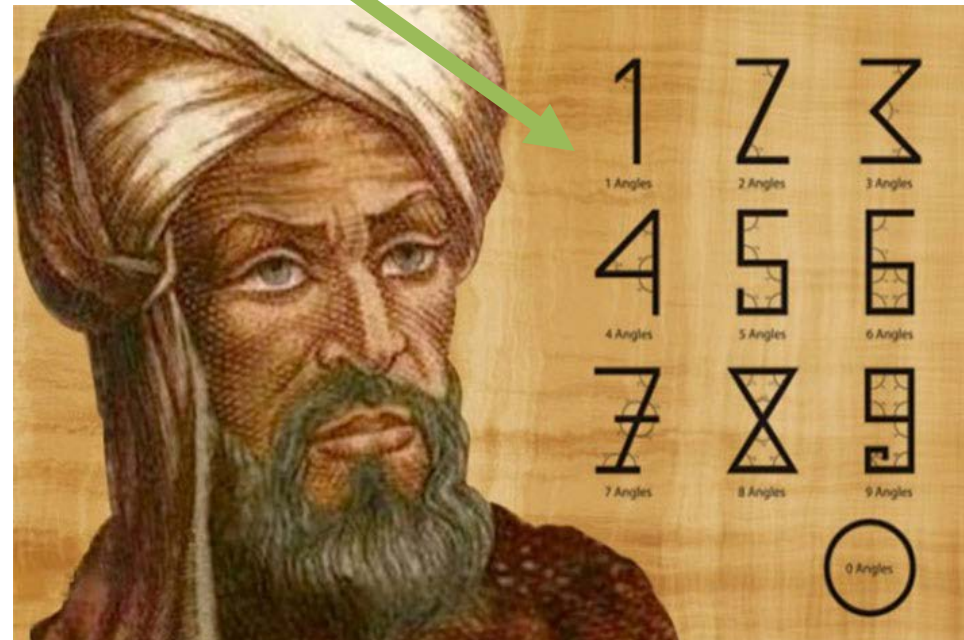
Introduced **Hindu-Arabic numerals** and the concepts of **algebra** into European mathematics.

Latinized versions of his
name and of his most
famous book title

(...*al-jabr*...)

live on in the terms
algorithm

and **algebra**.



Al-Khwārizmī

Muḥammad ibn Mūsā al-Khwārizmī
(circa 780-850),
Muslim mathematician and

Introduced **Hindu-Arabic
algebra** into European mat

Latinized versions of his
name and of his most
famous book title

(...*al-jabr*...)

live on in the terms
algorithm

and *algebra*.



(Wikipedia)

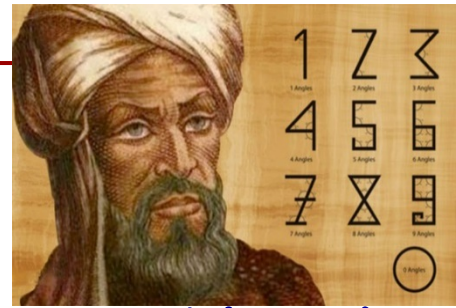
Algebra

- "reunion of broken parts"
- the study of mathematical symbols and rules for manipulating them

Definizioni buone ...



dare una definizione e poi confrontarla con quelle della pagina successiva



Definizioni buone ...

- una **sequenza di istruzioni** (che puo` essere applicata per risolvere un problema)
- una **procedura computazionale** che prende alcuni valori (**INPUT**) e produce (restituisce) dei valori (**OUTPUT**), applicando delle regole operative descritte nella procedura
- un **procedimento**, costituito da una **sequenza di passi**, applicata per risolvere un problema

... scritto in linguaggio naturale

(o in qualcosa di simile chiamato *pseudocodice*)

... **come sequenza di punti**
... **ordinatamente**



Un insieme di istruzioni

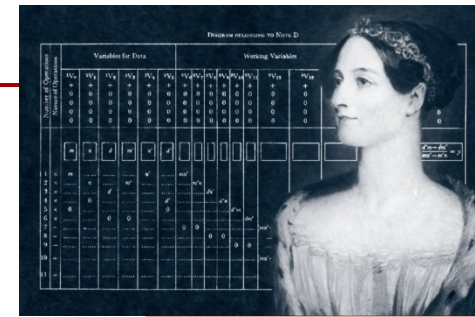
che descrive un algoritmo, lo traduce, in un linguaggio particolare

Il calcolatore esegue il programma ... nel senso che

- esegue la sequenza delle istruzioni ... nel loro ordine
 - riceve dei dati di input,
 - produce/calcola dati intermedi
 - produce/calcola dati di output,
- con lo scopo di risolvere un problema

Programma (eseguibile)

Un insieme di istruzioni macchina che descrive un algoritmo



(Ada Lovelace)

Il calcolatore esegue il programma ... nel senso che

- esegue la sequenza delle istruzioni ... nel loro ordine
 - riceve dei dati di input,
 - produce/calcola dati intermedi
 - produce/calcola dati di output,
- con lo scopo di risolvere un problema

istruzioni? What's istruzioni?

Un'istruzione è un comando che può essere eseguito dal calcolatore.

Ogni calcolatore (... , la sua CPU, o processore) ha un insieme di "istruzioni eseguibili dai suoi circuiti" (memorizzare dati, consultare dati, fare operazioni con i dati)

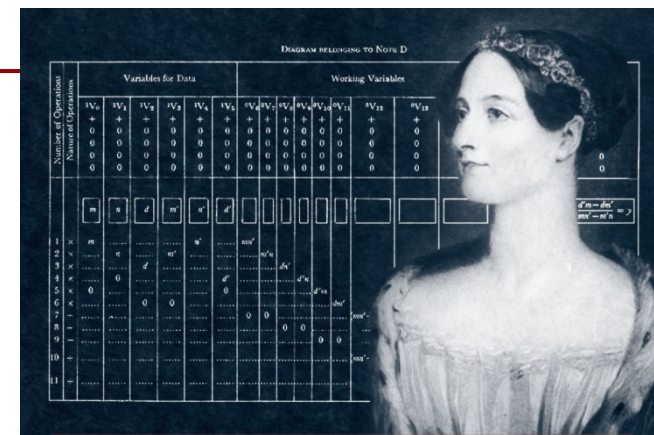
Sono le **istruzioni macchina**. Le uniche davvero eseguibili dalla CPU ...

Programma (in generale)

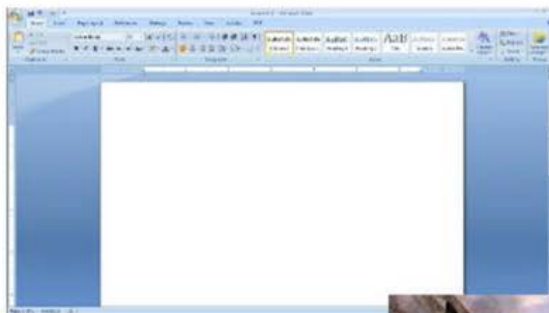
Un insieme di istruzioni
che viene passato ad un calcolatore

eseguendo le istruzioni del programma
il calcolatore

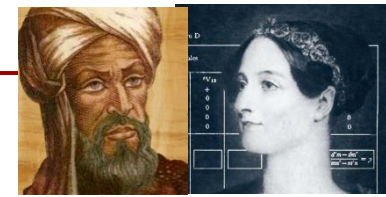
- esegue un algoritmo, cioè
 - processa dei dati di input,
 - produce dei dati di output,
- con lo scopo di risolvere un problema



Ada Lovelace day: Oct. 8th



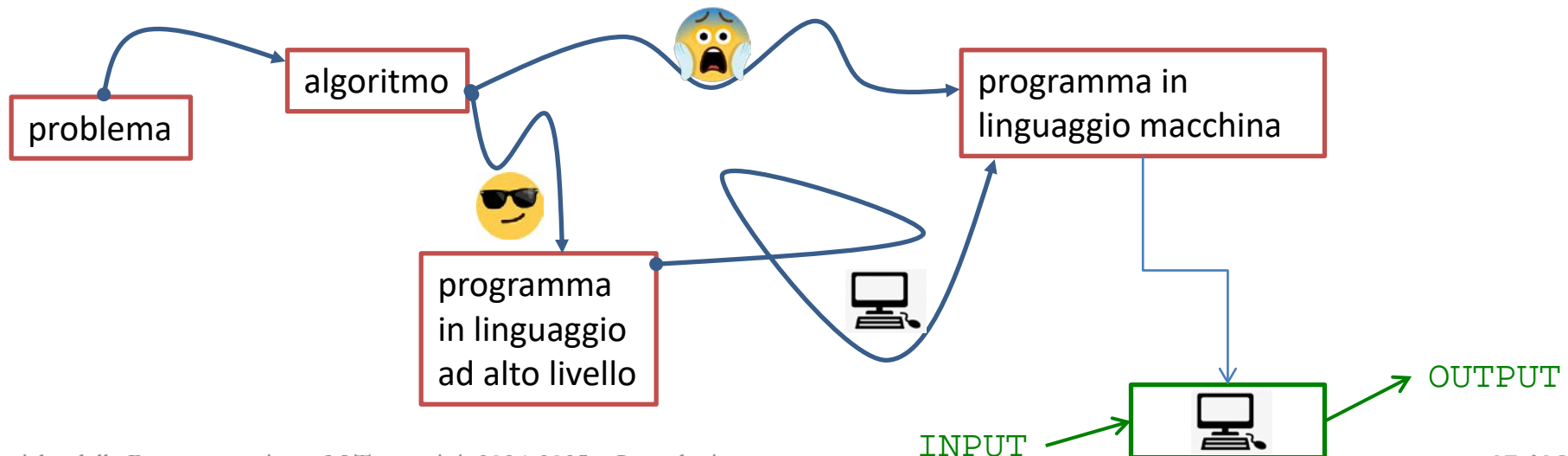
Algoritmo e Programma



Un **programma** e` scritto in un **linguaggio di programmazione**,
cioe` con regole grammaticali che permettono di scrivere la
sequenza di istruzioni in modo **comprensibile al calcolatore**

- o **direttamente** comprensibile (programma scritto direttamente nel linguaggio delle istruzioni macchina)
- o **indirettamente** (programma scritto in linguaggio ad alto livello e poi tradotto nel linguaggio delle istruzioni macchina)

Un **algoritmo** e` scritto in **linguaggio naturale**: anche se si cerca di essere il piu` precisi possibili, una frase scritta in linguaggio naturale puo` essere troppo complessa e ambigua per il calcolatore (non si puo` chiedere al calcolatore di eseguirlo direttamente ... e` per questo che l'algoritmo va traslato in un programma).



Tecniche della Programmazione, lez. 1

- Qualche informazione sul corso, il Programma
- Informatica, Calcolatore Elettronico, Algoritmo ...
- and now

Introduzione allo sviluppo ed esecuzione di programmi

Programmazione?

E' la stesura di una

certa sequenza di ISTRUZIONI, da far eseguire al calcolatore

(quindi comprensibili al calcolatore ...)

per risolvere un problema

Problema

Calcolare l'area di un
triangolo, considerando
base e altezza

Soluzione (tra noi eletta schiera)

faccio base per altezza e divido
per 2

Soluzione (per un bimbo)

vedi quant'e' la base

vedi quant'e' l'altezza

fai base per altezza

ora dividi per due

quello e'

come prima, ma piu' precisa, con meno
sottintesi

per un calcolatore?

serve ancora piu' precisa

Programmazione. In che senso??

determinare ed esprimere

STRUTTURA DATI

rappresentazione degli
oggetti / aspetti concreti
del problema

Algoritmo

Pianificazione azioni da
intraprendere sui dati (o
con i dati)

Programma C

POI traduzione delle azioni in istruzioni comprensibili (o meglio, che possono diventare comprensibili) al calcolatore



Nel nostro caso ... area del triangolo **DATI = ?**

Algoritmo ?

Programmazione. In che senso??

determinare ed esprimere

STRUTTURA DATI

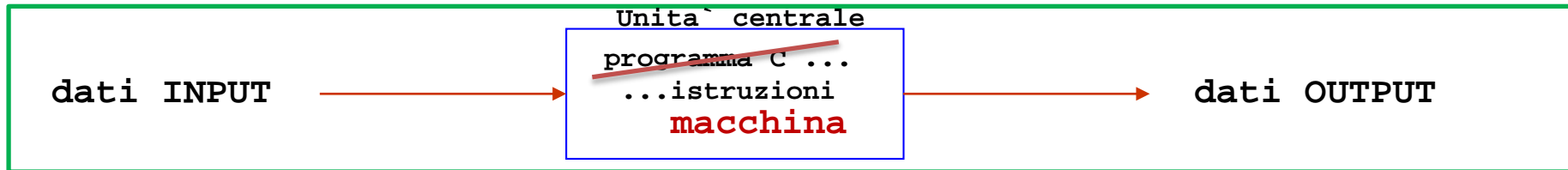
rappresentazione degli
oggetti / aspetti concreti
del problema

Algoritmo

Pianificazione azioni da
intraprendere sui dati (o
con i dati)

Programma C

traduzione delle azioni in istruzioni comprensibili (o meglio, che possono diventare comprensibili)
al calcolatore



DATI = base, altezza, area, 2

Algoritmo

ricevi base	(I)
ricevi altezza	(I)
area \leftarrow base*altezza/2	
emissione di area	(O)

Programmazione. In che senso??

determinare ed esprimere

STRUTTURA DATI

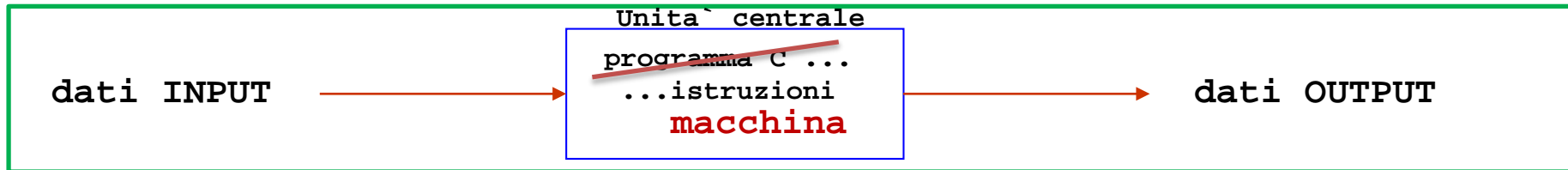
rappresentazione degli
oggetti / aspetti concreti
del problema

Algoritmo

Pianificazione azioni da
intraprendere sui dati (o
con i dati)

Programma C

traduzione delle azioni in istruzioni comprensibili (o meglio, che possono diventare comprensibili)
al calcolatore



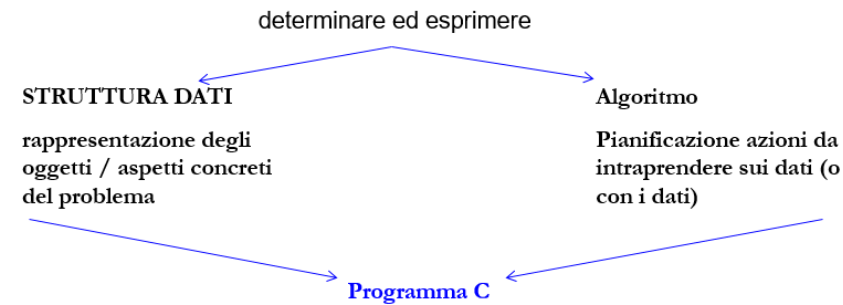
DATI = base, altezza, area, 2

Ora che abbiamo l'algoritmo, ,scriviamo il programma;
nel programma i dati saranno chiamato b,h,a e 2

Algoritmo ricevi base (I)
 ricevi altezza (I)
 $\text{area} \leftarrow \text{base} * \text{altezza} / 2$
 "**output**" di area (O)

Programma
 scanf ... b (I)
 scanf ... h (I)
 $a = b * h / 2$
 printf ... a (O)

Programmazione? Nel senso che ...



DATI = base, altezza, area, 2

Algoritmo

<u>ricevi base</u>	(I)
<u>ricevi altezza</u>	(I)
$\text{area} \leftarrow \text{base} * \text{altezza} / 2$	
"output" di area	(O)

prima si lavora per scrivere il programma in ... C ...

```
#include<stdio.h>

int main() {
    double b, h;
    double area;
    double prod;
    scanf("%lf %lf", &b, &h);
    prod = b*h;
    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

    return 0;
}
```

Poi il programma viene tradotto in istruzioni macchina e puo` quindi essere eseguito al calcolatore

Dati e STRUTTURA DATI

DATI =

valori numerici ...

o insiemi coerenti di valori numerici

associati a informazioni significative per il programma (e il programmatore)

- base, altezza ... naturalmente numeri
- lettera Z? (in effetti corrisponde ad un numero, un codice in una tabella ...)
- persona = {nome, cognome, eta`, peso, data_nascita, Codice_Fiscale}
= gruppi di caratteri, numeri, cifre intervallate da '/' ,

il lavoro di definire
quali dati corrispondono a quali informazioni,
e come tali dati sono conformati,
e` la *"definizione delle strutture dati"*
cruciale in un programma!!

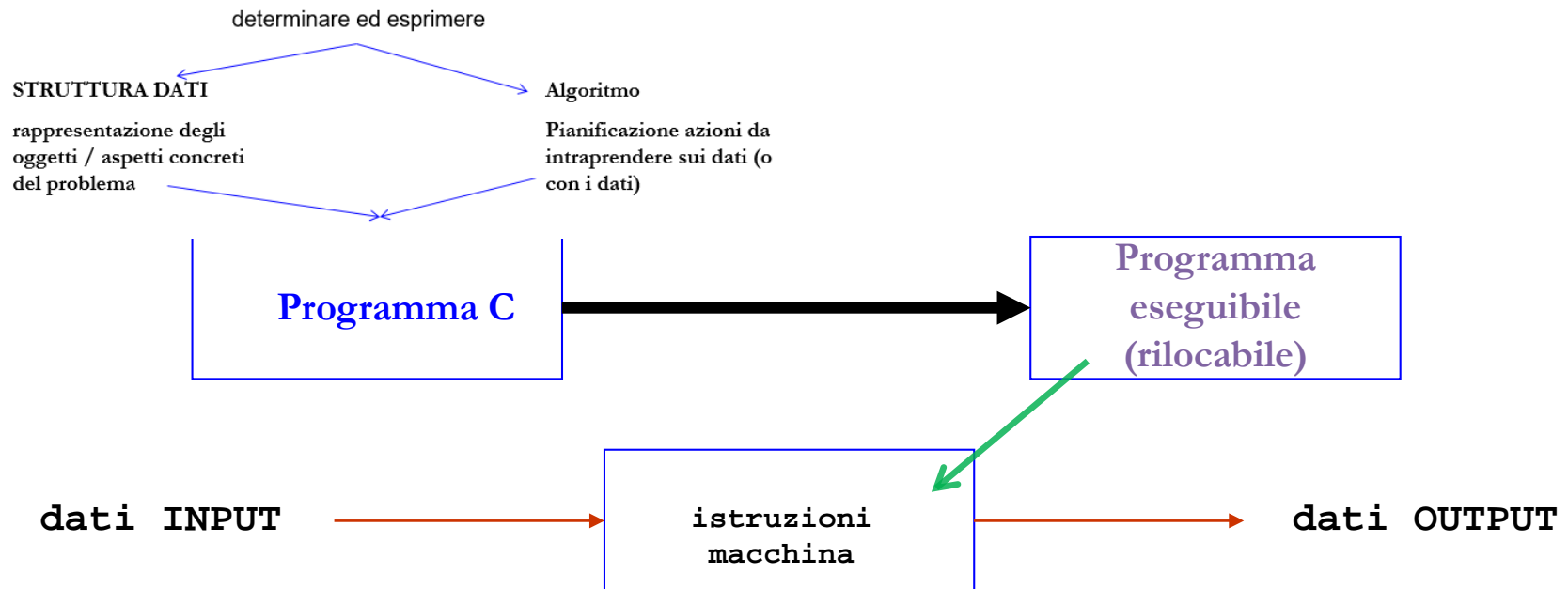
Il calcolatore **esegue** il programma; il programmatore **programma**

CALCOLATORE =

eseguire istruzioni per
ricevere / ricordare / calcolare
dati

PROGRAMMATORE =

Definire **RAPPRESENTAZIONE DATI**
e programmare **AZIONI SU DATI**,
in modo comprensibile al calcolatore (o
che puo` diventare automaticamente
comprensibile al calcolatore)



Sviluppo di un programma

Metodologia elementare

- Specifica del problema
- Analisi del problema
- Sintesi algoritmo
- Programmazione algoritmo
(con *codifica* in un linguaggio)



Esempio

- **specifica** del problema "area"

-ANALISI: QUALI dati? Come FATTI (struttura)?

- INPUT b,h numeri reali (double)

- OUTPUT area numero reale

- dati per calcoli intermedi ... prod

- idea: prod ← b*h

 area ← prod/2

SINTESI ... =

- 1) prendere da INPUT i valori da associare a **b** e **h**
- 2) calcolare **prod**
- 3) calcolare **area**
- 4) fornire in OUTPUT il valore di **area**

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $\text{prod} \leftarrow b \cdot h$
 $\text{area} \leftarrow \text{prod} / 2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT** i valori da associare a b e h

2) calcolare prod

3) calcolare area

4) fornire in **OUTPUT** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {

    double b, h;

    double area;

    double prod;

    scanf("%lf %lf", &b, &h);

    prod = b*h;

    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

    return 0;

}
```

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $prod \leftarrow b \cdot h$
 $area \leftarrow prod / 2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {

    double b, h;

    double area;

    double prod;

    scanf("%lf %lf", &b, &h);

    prod = b*h;

    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e ` %g\n", b, h,
           area);

    return 0;

}
```

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $prod \leftarrow b \cdot h$
 $area \leftarrow prod/2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>
```

```
int main() {
```

```
    double b, h;
```

```
    double area;
```

```
    double prod;
```

```
    scanf("%lf %lf", &b, &h);
```

```
    prod = b*h;
```

```
    area = prod/2;
```

```
    printf("il valore dell'area di un  
          triangolo avente base = %g e  
          altezza = %g e ` %g\n", b, h,  
          area);
```

```
    return 0;
```

```
}
```


Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $prod \leftarrow b \cdot h$
 $area \leftarrow prod/2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>
```

```
int main() {
```

```
    double b, h;
```

```
    double area;
```

```
    double prod;
```

```
    scanf("%lf %lf", &b, &h);
```

1

```
    prod = b*h;
```

```
    area = prod/2;
```

```
    printf("il valore dell'area di un  
          triangolo avente base = %g e  
          altezza = %g e` %g\n", b, h,  
          area);
```

```
    return 0;
```

```
}
```

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $\text{prod} \leftarrow b \cdot h$
 $\text{area} \leftarrow \text{prod} / 2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {

    double b, h;

    double area;

    double prod;

    scanf("%lf %lf", &b, &h);

    prod = b*h;

    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

    return 0;

}
```

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $prod \leftarrow b \cdot h$
 $area \leftarrow prod/2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {

    double b, h;

    double area;

    double prod;

    scanf("%lf %lf", &b, &h);

    prod = b*h;

    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e ` %g\n", b, h,
           area);

    return 0;

}
```

Sviluppo di un programma

-ANALISI:

QUALI dati? FATTI come (struttura)?

- INPUT b,h numeri reali (double)
- OUTPUT area numero reale
- dati per calcoli intermedi ...
- idea: $prod \leftarrow b \cdot h$
 $area \leftarrow prod / 2$

SINTESI ... = ALGORITMO

0) informazioni rappresentate da b per la base, h per l'altezza, a per l'area

1) prendere da **INPUT (LETTURA, ACCESSO e ASSEGNAZIONE/MEMORIZZAZIONE)** i valori da associare a b e h

2) calcolare prod (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

3) calcolare area (**ACCESSO, CALCOLO, ASSEGNAZIONE/MEMORIZZAZIONE**)

4) fornire in **OUTPUT (ACCESSO e RITTURA)** il valore di area

4

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {

    double b, h;

    double area;

    double prod;

    scanf("%lf %lf", &b, &h);

    prod = b*h;

    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

return 0;

}
```

4

esecuzione? di un programma

Assumiamo che il calcolatore abbia la traduzione del programma in istruzioni macchina e simuliamo l'esecuzione del programma ...

PROGRAMMAZIONE ... =

```
#include<stdio.h>

int main() {
    double b, h;
    double area;
    double prod;
    scanf("%lf %lf", &b, &h);
    prod = b*h;
    area = prod/2;
    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e ` %g\n", b, h,
           area);
    return 0;
}
```

INPUT

traduzione del
programma C in
istruzioni macchina

OUTPUT



MEMO
RIA
CENTR
ALE
(circa)

address

"locazione"

esecuzione? di un programma

PROGRAMMAZIONE ... =

```

#include<stdio.h>

int main() {
    double b, h; <-----
    double area; <-----
    double prod; <-----

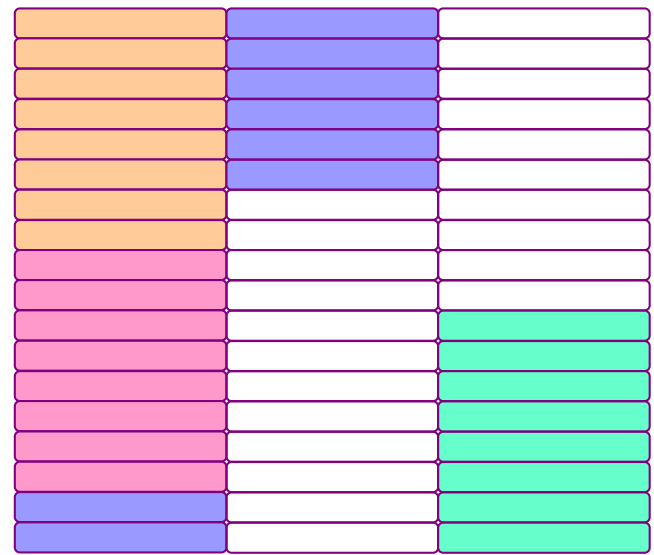
    scanf("%lf %lf", &b, &h);

    prod = b*h;

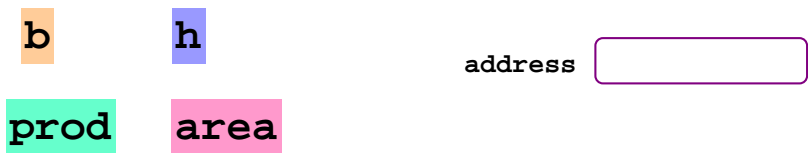
    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

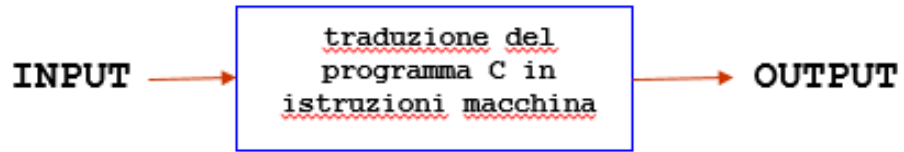
    return 0;
}
    
```



MEMORIA
(circa)



"dichiarazione di variabile"
 "variabile" =
 identificatore, indirizzo, contenuto



esecuzione? di un programma

PROGRAMMAZIONE ... =

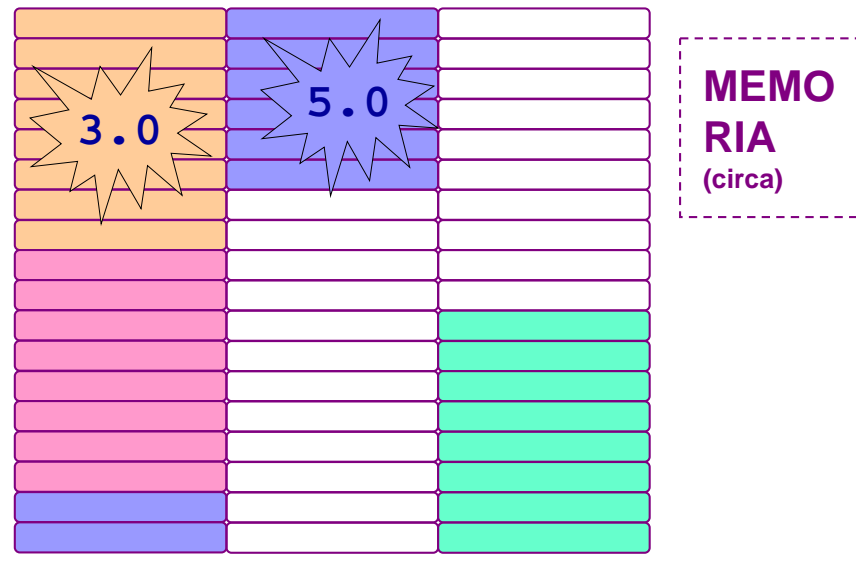
```
#include<stdio.h>

int main() {
    double b, h;
    double area;
    double prod;

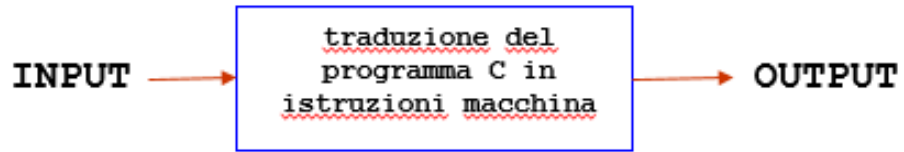
    scanf("%lf %lf", &b, &h); <-----
    prod = b*h;
    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);

    return 0;
}
```



"variabile"
 identificatore, indirizzo, contenuto



esecuzione? di un programma

PROGRAMMAZIONE ... =

```

#include<stdio.h>

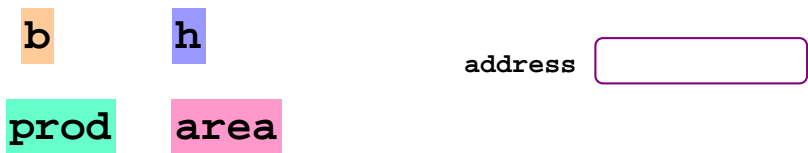
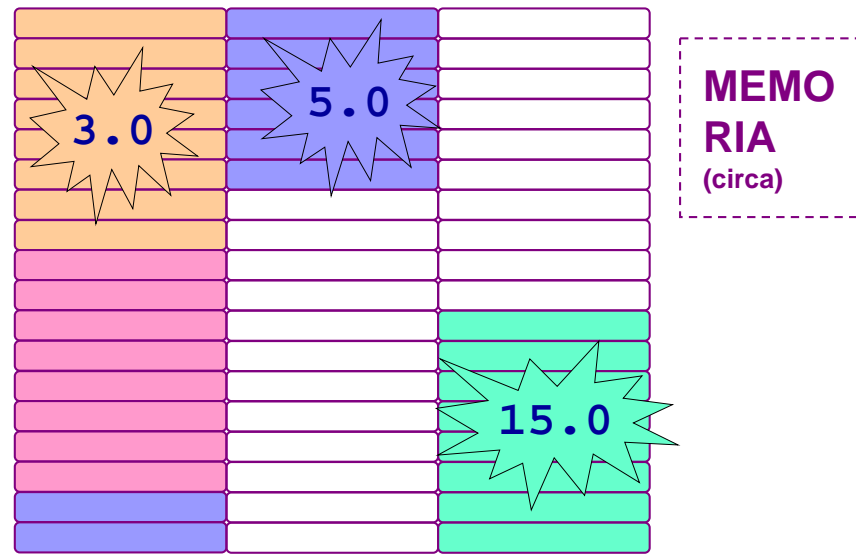
int main() {
    double b, h;
    double area;
    double prod;

    scanf("%lf %lf", &b, &h);

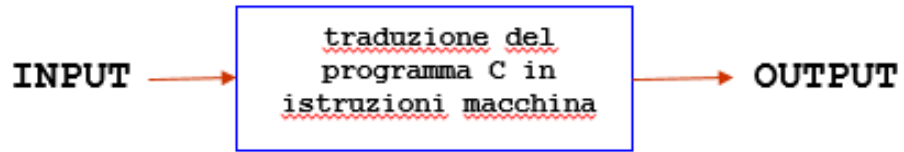
    prod = b*h;
    area = prod/2;

    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e ` %g\n", b, h,
           area);

    return 0;
}
    
```



"variabile"
 identificatore, indirizzo, contenuto



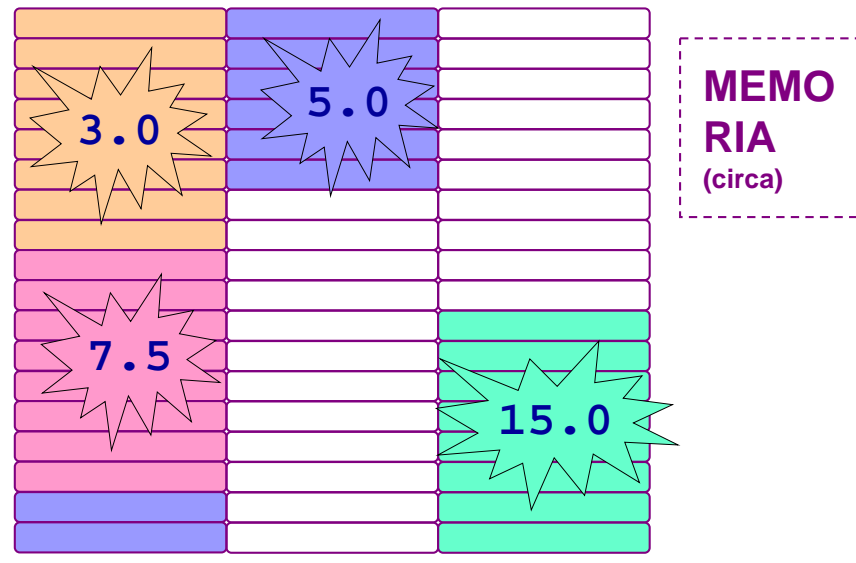
esecuzione? di un programma

PROGRAMMAZIONE ... =

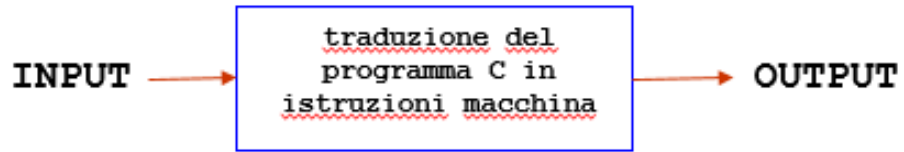
```

#include<stdio.h>

int main() {
    double b, h;
    double area;
    double prod;
    scanf("%lf %lf", &b, &h);
    prod = b*h;
    area = prod/2;
    printf("il valore dell'area di un
           triangolo avente base = %g e
           altezza = %g e` %g\n", b, h,
           area);
    return 0;
}
    
```



"variabile"
 identificatore, indirizzo, contenuto



esecuzione? di un programma

PROGRAMMAZIONE ... =

```
#include<stdio.h>

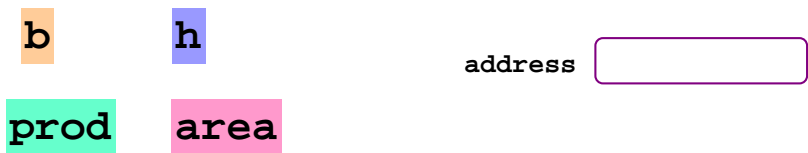
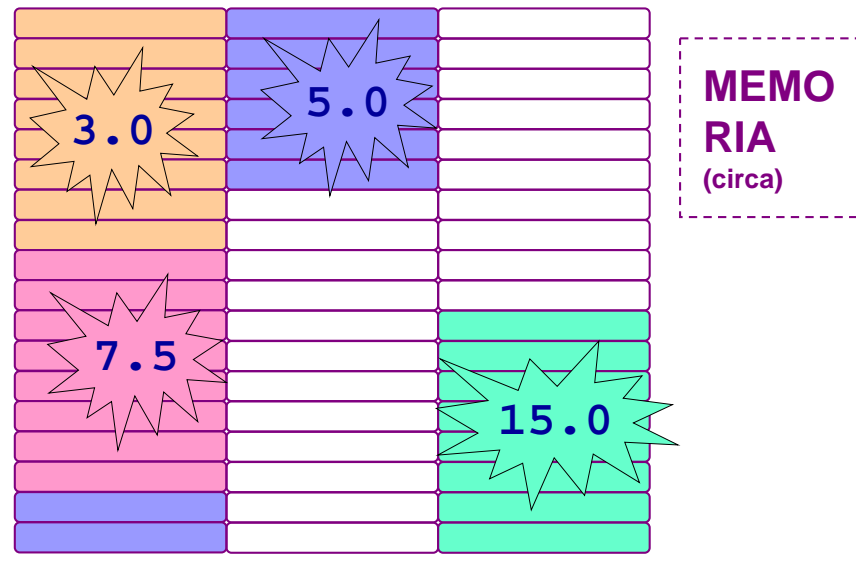
int main() {
    double b, h;
    double area;
    double prod;

    scanf("%lf %lf", &b, &h);

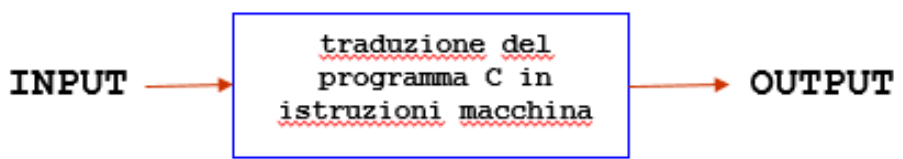
    prod = b*h;
    area = prod/2;

    printf("il valore dell'area di un
        triangolo avente base = %g e
        altezza = %g e` %g\n", b, h,
        area);

return 0;
}
```



"variabile"
 identificatore, indirizzo, contenuto



bla bla ...
 base = 3 e altezza = 5
 bla bla 7.5

Tecniche della Programmazione, lez. 1

- soluzioni

processo per arrivare a risolvere un problema con un programma

a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma

ESERCIZIO ... la soluzione e' nelle slide seguenti: non guardarle subito!



processo per arrivare a risolvere un problema con un programma

a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma



a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma

PROGRAMMA

↓
TROVI UNA PROCEDURA CHE
TI PERMETTE DI RISOLVERE
IL PROBLEMA (SU OGNI
POSSIBILE INSIEME DI DATI)

TRASFORMI LA PROCEDURA
IN UNA SEQUENZA DI OPERAZIONI
CHE POSSONO ESSERE ESEGUITE
DAL CALCOLATORE

SE VUOI RISOLVERE IL
PROBLEMA SU UN CERTO
INSIEME DI DATI

RAPPRESENTI I DATI NEL
CALCOLATORE E FAI ESEGUIRE AL
CALCOLATORE LA SEQUENZA DI
OPERAZIONI CHE PERMETTE DI
RISOLVERE IL PROBLEMA

a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma

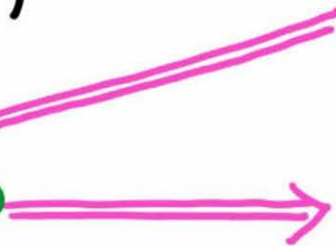


TROVI UNA PROCEDURA CHE
TI PERMETTE DI RISOLVERE
IL PROBLEMA (SU OGNI
POSSIBILE INSIEME DI DATI)



TRASFORMI LA PROCEDURA
IN UNA SEQUENZA DI OPERAZIONI
CHE POSSONO ESSERE ESEGUITE
DAL CALCOLATORE

SE VUOI RISOLVERE IL
PROBLEMA SU UN CERTO
INSIEME DI DATI
↓
ISTANZA



RAPPRESENTI I DATI NEL
CALCOLATORE E FAI ESEGUIRE AL
CALCOLATORE LA SEQUENZA DI
OPERAZIONI CHE PERMETTE DI
RISOLVERE IL PROBLEMA

a quali blocchi associare le seguenti parole?

- esecuzione
- algoritmo
- istanza
- programma



TROVI UNA PROCEDURA CHE
TI PERMETTE DI RISOLVERE
IL PROBLEMA (SU OGNI
POSSIBILE INSIEME DI DATI)

TRASFORMI LA PROCEDURA
IN UNA SEQUENZA DI OPERAZIONI
CHE POSSONO ESSERE ESEGUITE
DAL CALCOLATORE

SE VUOI RISOLVERE IL
PROBLEMA SU UN CERTO
INSIEME DI DATI

RAPPRESENTI I DATI NEL
CALCOLATORE E FAI ESEGUIRE AL
CALCOLATORE LA SEQUENZA DI
OPERAZIONI CHE PERMETTE DI
RISOLVERE IL PROBLEMA

ESECUZIONE
DEL PROGRAMMA

riempire il vuoto (e solo poi guardare la prossima slide)

PROBLEMI, ALGORITMI, PROGRAMMI

PROBLEMI DI INTERESSE SONO QUELLI PER I QUALI È POSSIBILE UNA
formalizzazione matematica

SI POSSONO RISOLVERE ATTRAVERSO UN algoritmo, CHE CONSISTE NELLA
sequenza di passi operativi CHE CONSENTE DI OTTENERE
i risultati dell'istanza del problema, a partire dai dati di input

Si può incaricare un "esecutore automatico" di eseguire l'algoritmo, fornendogli una in un linguaggio di programmazione,
cioè fornendogli un programma

PROBLEMI, ALGORITMI, PROGRAMMI

PROBLEMI DI INTERESSE SONO QUELLI PER I QUALI È POSSIBILE UNA
formalizzazione matematica

SI POSSONO RISOLVERE ATTRAVERSO UN algoritmo, CHE CONSISTE NELLA
sequenza di passi operativi CHE CONSENTE DI OTTENERE
i risultati dell'istanza del problema, a partire dai dati di input

Si può incaricare un "esecutore automatico" di eseguire l'algoritmo, fornendogli una traduzione dell'algoritmo in un linguaggio di programmazione,
cioè fornendogli un programma