CdLM in Management Digitale

UNIVERSITÀ DEL SALENTO

Tecnologie Digitali

Dip.to di Scienze dell'Economia

Laurea Triennale

8 cfu

Prof. Salvatore Mancarella

salvatore.mancarella@unisalento.it

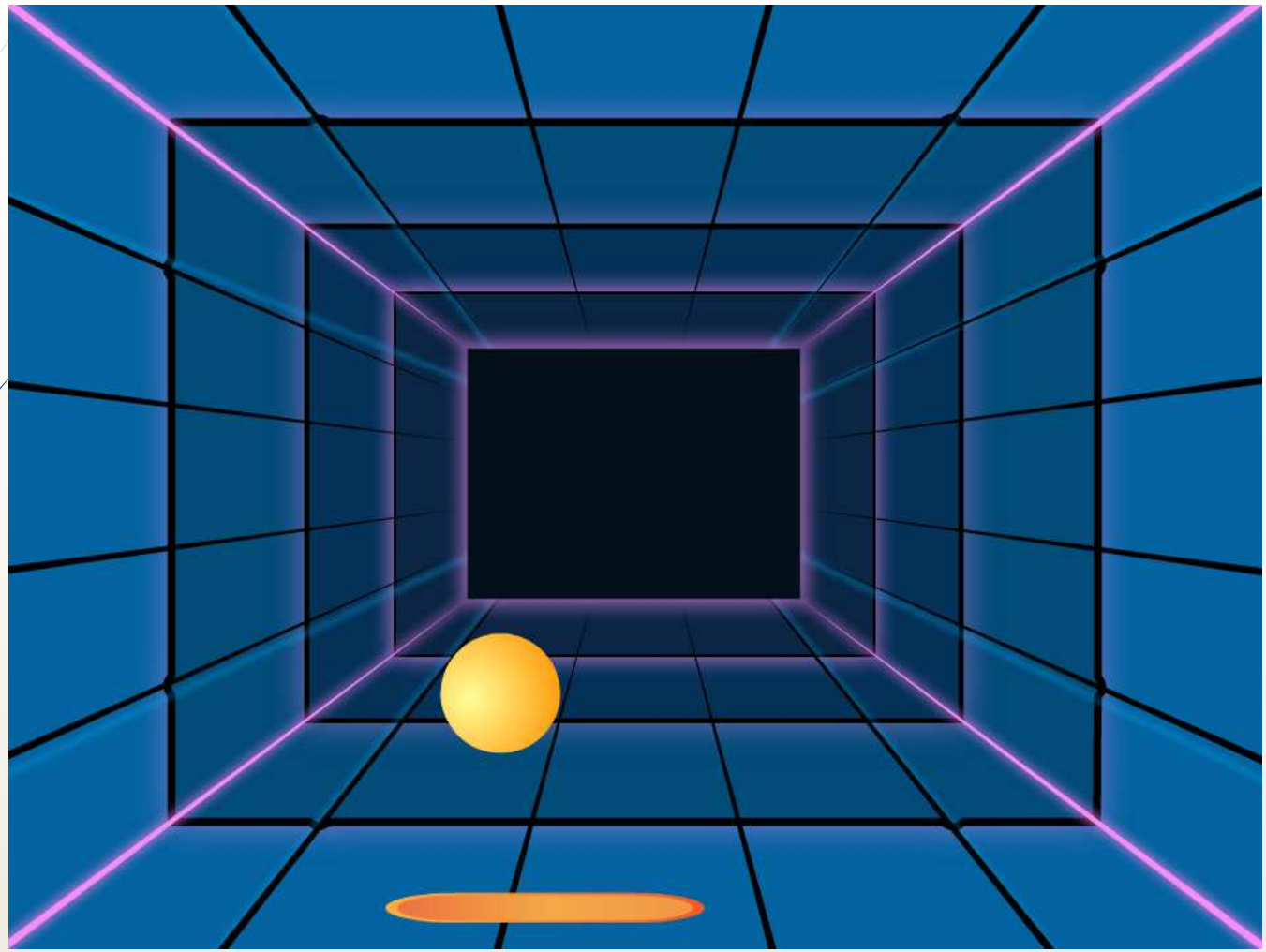
Pensiero computazionale

Sommario

- ✓Pensiero computazionale
- ✓Coding
- ✓Algoritmi
- ✓Diagramma a blocchi

Tre domannde

- Cos'e il pensiero computazionale?
- Cos'è il coding?
- Infine, cosa c'entra questo con la vostra vita professionale



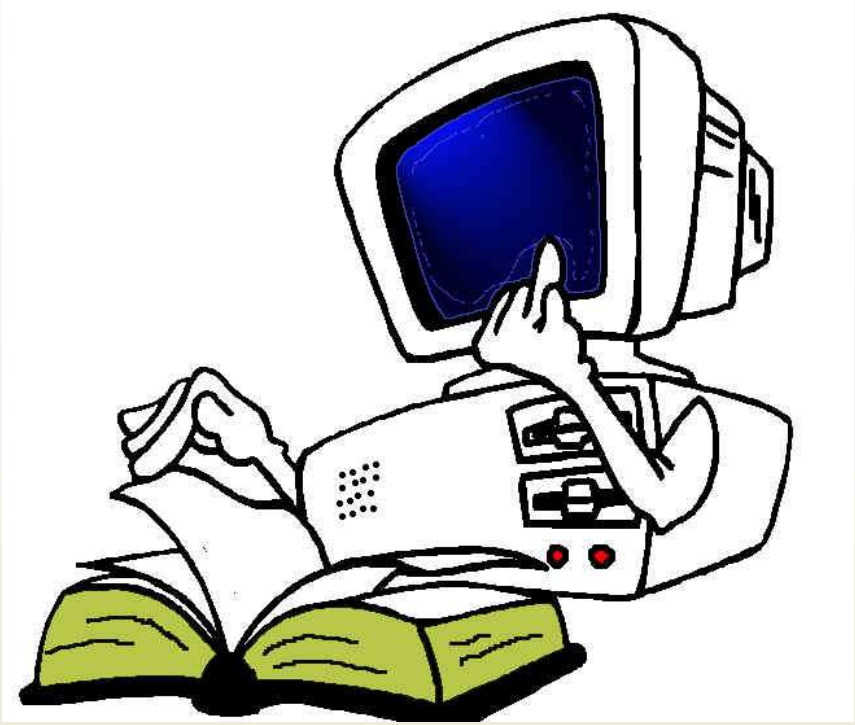
Stress

Stress - Soluzione

Utilizzare il pensiero computazionale

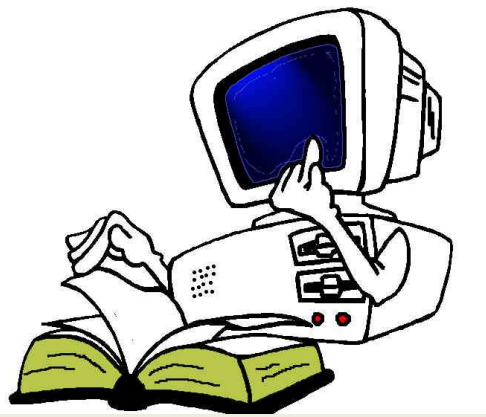
IL pensiero computazionale

Che cos'è?



Il pensiero computazionale

E' un modo di pensare che ci aiuta a formulare «soluzioni costruttive» a problemi che dobbiamo risolvere



Facciamo un esempio

Se vuoi raggiungere una vetta….

non puoi pensare di…

Arrivarci subito e senza allenamento

L'obiettivo

va raggiunto a piccoli passi.

Il pensiero computazionale è lo strumento che ci consente di passare da un'idea al procedimento per realizzarla

Imparare a programmare è il modo migliore per acquisire il pensiero computazionale

Quindi se la meta è la vetta

devi allenarti

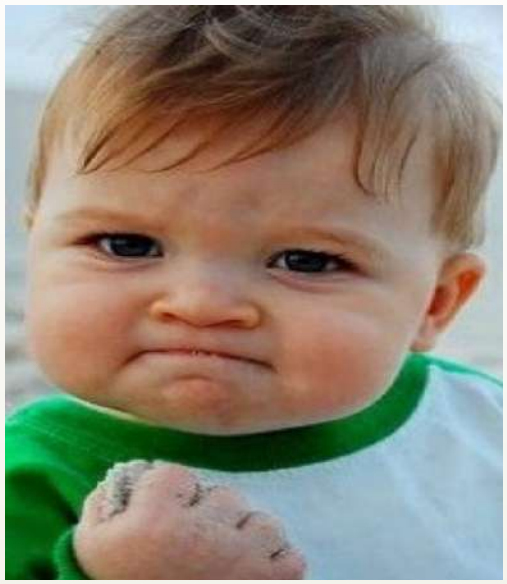
Pensiero computazionale

Definizione formulata dalla dottoressa Jeannette Wing, direttrice del Dipartimento di Informatica della Carnegie Mellon University, secondo cui

« il processo mentale che sta alla base della formulazione dei problemi e delle loro soluzioni così che le soluzioni siano rappresentate in una forma che può essere implementata in maniera efficace da un elaboratore di informazioni sia esso umano o artificiale».

Pensiero computazionale

Ovvero è lo sforzo che un individuo deve mettere in atto per fornire a un altro individuo o macchina tutte e sole le "istruzioni" necessarie affinché questi eseguendole sia in grado di portare a termine il compito dato.



Pensiero computazionale

DETTO IN PAROLE SEMPLICI

QUANDO AFFRONTIAMO UN PROBLEMA O ABBIAMO UN'IDEA, SPESSO INTUIAMO LA SOLUZIONE MA NON SIAMO IN GRADO DI FORMULARLA IN MODO OPERATIVO PER METTERLA IN PRATICA.

Il PENSIERO COMPUTAZIONALE E' QUESTO

LA CAPACITÀ DI DESCRIVERE UN PROCEDIMENTO COSTRUTTIVO CHE PORTI AD UNA SOLUZIONE CREATIVA, EFFICACE E NON AMBIGUA

“Quando si diventa fluenti a leggere e scrivere non lo si fa solamente per diventare uno scrittore di professione.

Ma imparare a leggere e scrivere è utile a tutti. Ed è la stessa cosa per la programmazione. La maggior parte delle persone non diventerà un esperto di informatica o un programmatore, ma l’abilità di pensare in modo creativo, pensare schematicamente, lavorare collaborando con gli altri […] sono cose che le persone possono usare, indipendentemente dal lavoro che fanno”.

Mitchel Resnick

Pensiero computazionale

pensiero computazionale,
aiuta a sviluppare competenze logiche e capacità di risolvere problemi in modo creativo ed efficiente, qualità che sono importanti per tutti gli ambiti lavorativo e sociale.

Il modo più semplice e divertente di sviluppare il pensiero computazionale è attraverso la programmazione (**coding**) in un contesto di gioco.

Coding

Cosa è il coding?

- Per coding si intende, in informatica, la stesura di un programma, cioè di una di quelle sequenze di istruzioni che, eseguite da un calcolatore, danno vita alla maggior parte delle meraviglie digitali che usiamo quotidianamente
- Il coding fornisce una forma mentis che permetterà loro di affrontare problemi complessi
- Insomma imparare a programmare apre le mente

Coding, codifica, codice…

L'atto in cui si scrive codice utile a definire comandi che una macchina eseguirà.

La nostra vita quotidiana è caratterizzata da codici, da linguaggi di programmazione utili a compiere azioni come:

- Leggere la posta elettronica
- Interagire con Facebook

...o anche

- Prelevare al bancomat
- Inviare un sms
- Acquistare un biglietto del bus da un dispositivo automatico

I linguaggi di programmazione sono definiti **da regole**. Saper programmare significa saper riconoscere quali regole sussistono affinché specifici **comandi** possano **generare il risultato atteso**.

OBAMA STANZIA 4 MILIARDI DI DOLLARI E PROMUOVE L'ORA DEL CODICE

IL 65% DEI BAMBINI CHE INIZIANO LA SCUOLA PRIMARIA FINIRANNO PER INTRAPRENDERE PROFESSIONI CHE OGGI ANCORA NON ESISTONO

fonte: The Future of Jobs and Skills (WEF 2016)

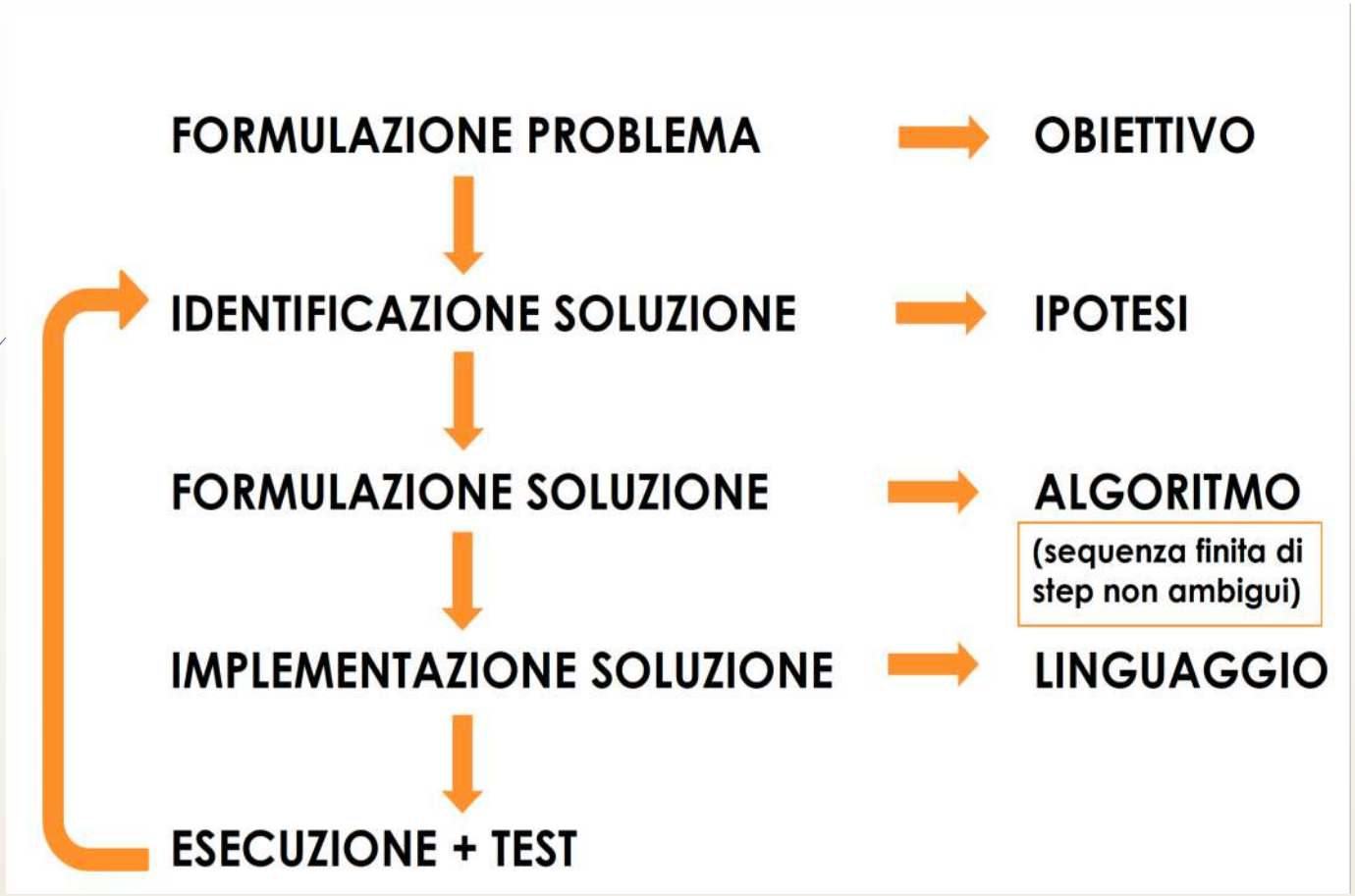
PENSARE COME UN INFORMATICO PER RISOLVERE PROBLEMI

SEARCHING TO SPEACK

Paul Curzon, Queen Mary University of London

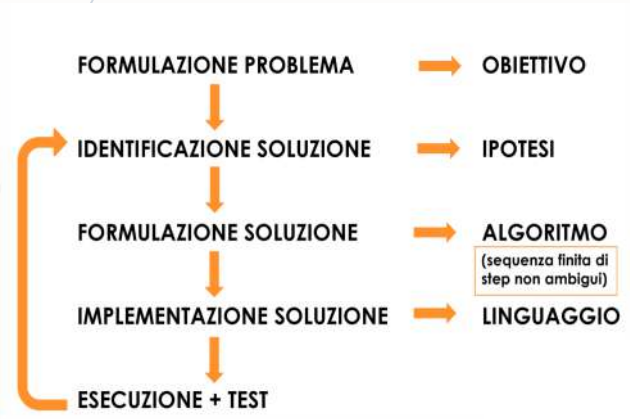
CODING
=
APPLICAZIONE INTUITIVA DI CONCETTI E PRATICHE DI PROGRAMMAZIONE FINALIZZATA ALLO SVILUPPO DEL PENSIERO COMPUTAZIONALE

Step pensiero computazionale

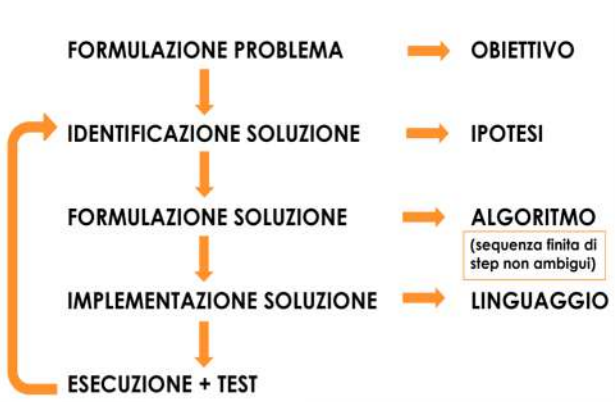


Scomposizione

Problema 1



Problema 2



pensiero computazionale

Algoritmi

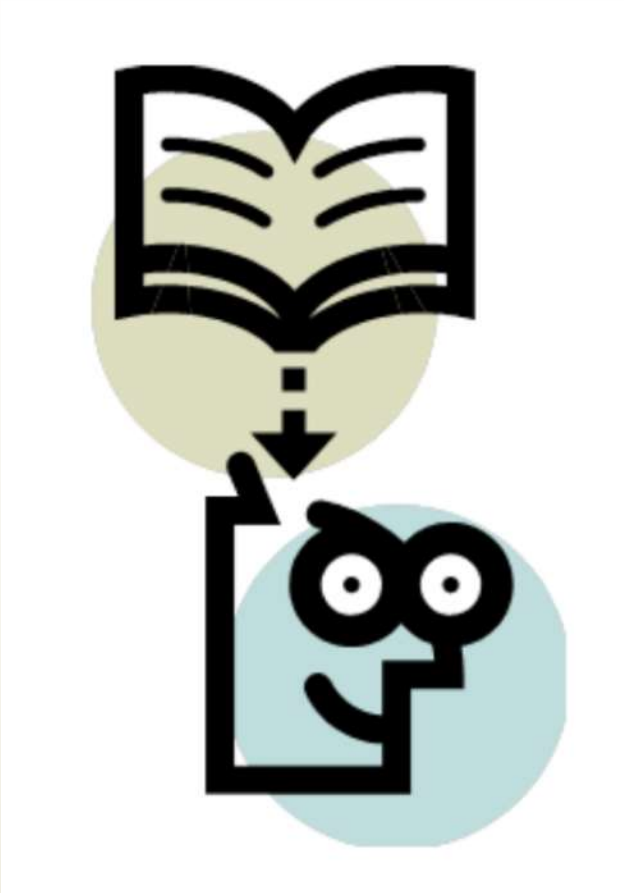
Algoritmi

Ogni giorno siamo chiamati a risolvere dei problemi, per es:

1. Decidere se riconciliarci con l'amico con cui abbiamo litigato;
2. Calcolare l'area di un rettangolo, note la base e l'altezza;
3. Stabilire quale squadra di calcio vincerà il campionato, ecc.

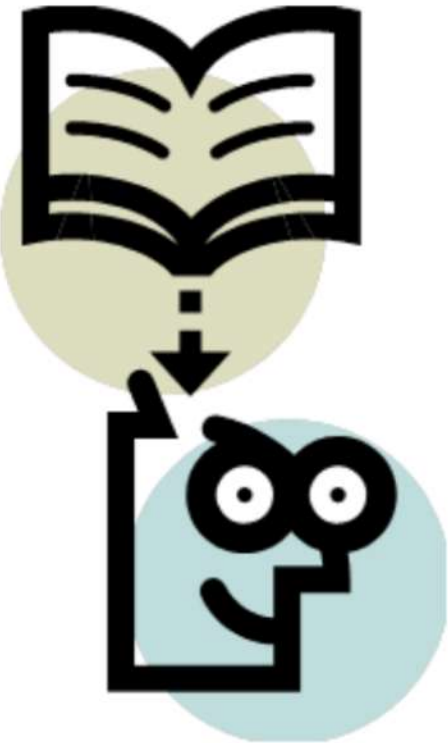
Noi cercheremo in questa sede di esaminare soltanto i problemi del secondo tipo, poiché solo tali problemi sono risolvibili mediante algoritmi

Cos'è un Algoritmi



Cos'è un Algoritmi

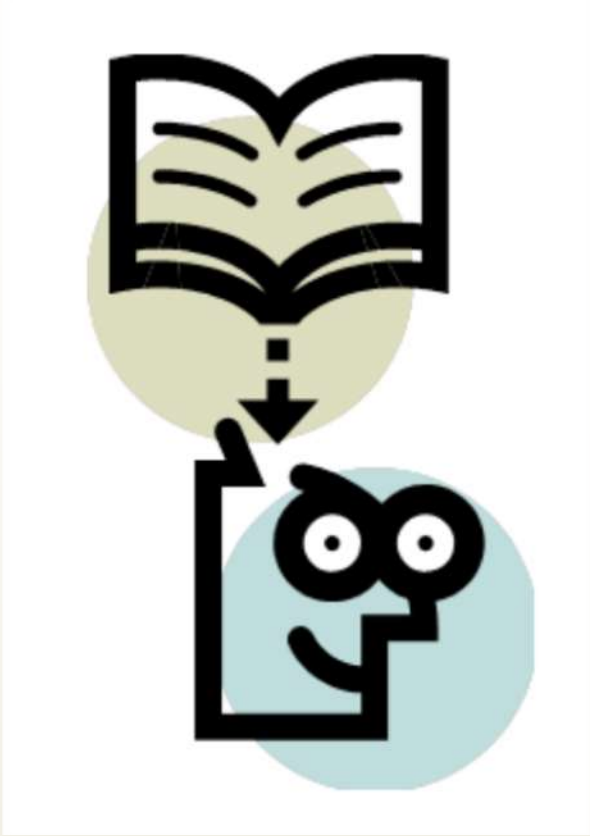
- E' una sequenza finita di passi (istruzioni), che devono essere eseguiti secondo un ordine prefissato per individuare la soluzione



Algoritmi

Quando dobbiamo risolvere un problema è necessario individuare:

- Chi lo risolve;
- Quali risorse sono necessarie;
- Qual è il risultato che si vuole ottenere



Algoritmi - Chi lo risolve

Es. devo spostarmi da un luogo ad un altro in comodità e velocemente

- Chi risolve questo problema

Algoritmi - Quali risorse sono necessarie

- Il carburante
- Il conducente
- …..

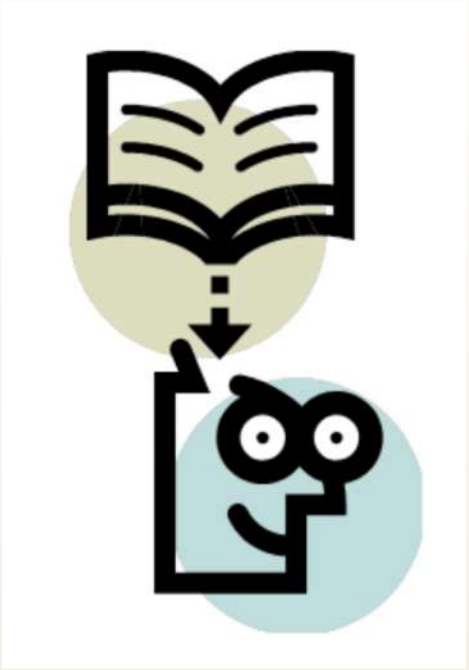
Algoritmi - Qual è il risultato che si vuole ottenere

- Arrivare a destinazione nel minor tempo possibile e comodamente

Algoritmi

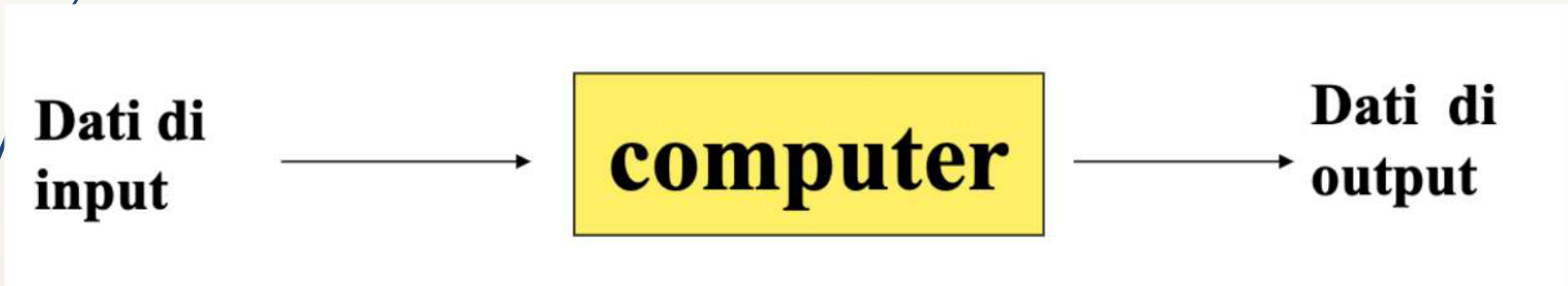
E' però necessario analizzare prima il problema e poi trovare una strategia per risolverlo.

- Mentre la macchina può essere utilizzata solo per spostarsi,
- il COMPUTER è in grado di risolvere problemi di natura diversa.



Cos'è il COMPUTER?

E' un sistema di elaborazione che trasforma le informazioni che riceve in ingresso nei risultati che vogliamo ottenere



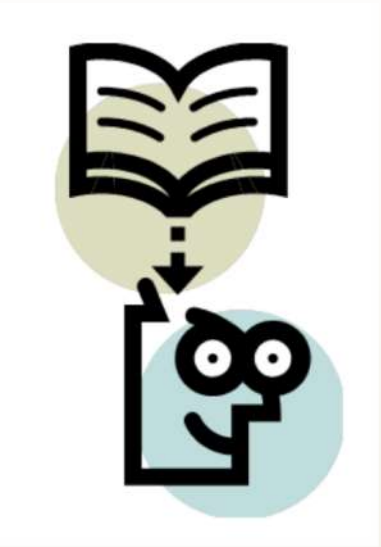
Caratteristiche dell'algoritmo

L'algoritmo deve essere:

- **Finito:** composto da un numero finito di passi
- **Deterministico:** a fronte degli stessi dati in input deve produrre gli stessi risultati
- **Non ambiguo:** i passi che lo compongono devono essere interpretati in modo univoco dall'esecutore
- **Generale:** deve fornire la soluzione per tutti i problemi appartenenti ad una certa classe

Componenti di un algoritmo

- **DATI**: sono gli oggetti su cui dobbiamo operare
- **ISTRUZIONI:** sono le attività che devono essere svolte



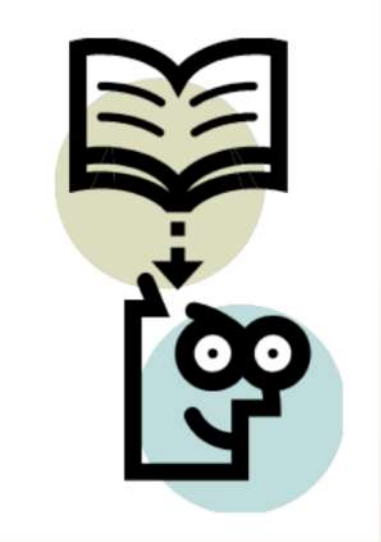
I dati

- Ad ogni dato è associato un nome che lo identifica in modo univoco (es. lato, codice, imposta, ecc).
- I dati possono essere classificati secondo il modo in cui interagiscono con l'elaboratore:
  - **Input** (forniti dall'esterno, es. prezzo, quantità, aliquota)
  - **Output** (comunicati all'esterno, es. totale fattura)
  - **Lavoro** (dati di lavoro, es. IVA).

I dati

A seconda degli oggetti che rappresentano, possono essere:

- **Numerici**
  - Interi (es: il numero di abitanti – 1,2,3..)
  - Reali (es: la temperatura 27,5° 28,6°)
- **Alfanumerici o stringhe**
  - Codice fiscale, indirizzo, ecc..



I dati

In base alla possibilità di cambiare il valore durante l'esecuzione dell'algoritmo, distinguiamo:

- **Costanti**, il cui valore rimane immutato nel tempo (es.aliquota IVA);
- **Variabili**, il cui valore può cambiare nel tempo (es. prezzo di un prodotto)

Le istruzioni

Le istruzioni Analizziamo i tre tipi di istruzioni fondamentali:

- **Lettura**: attraverso la quale si assegna ad una variabile un valore tramite digitazione sulla tastiera del PC;
- **Scrittura**: permette di visualizzare tramite video o supporto cartaceo, un messaggio o il valore di una variabile;
- **Assegnazione**: permette di attribuire un valore ad una variabile.
  - Es: A=3; oppure A=B+3;

Rappresentazione degli algoritmi

- **Diagramma a blocchi o di flusso o Flow-chart:** è il metodo più usato e si basa sull'uso di simboli a cui corrispondono delle precise operazioni.
- **Pseudocodifica** : utilizza un linguaggio speciale per descrivere le istruzioni da eseguire

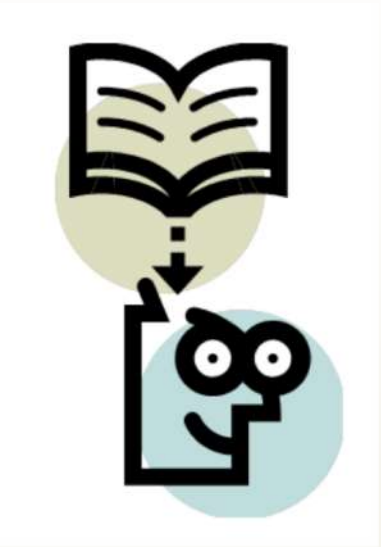
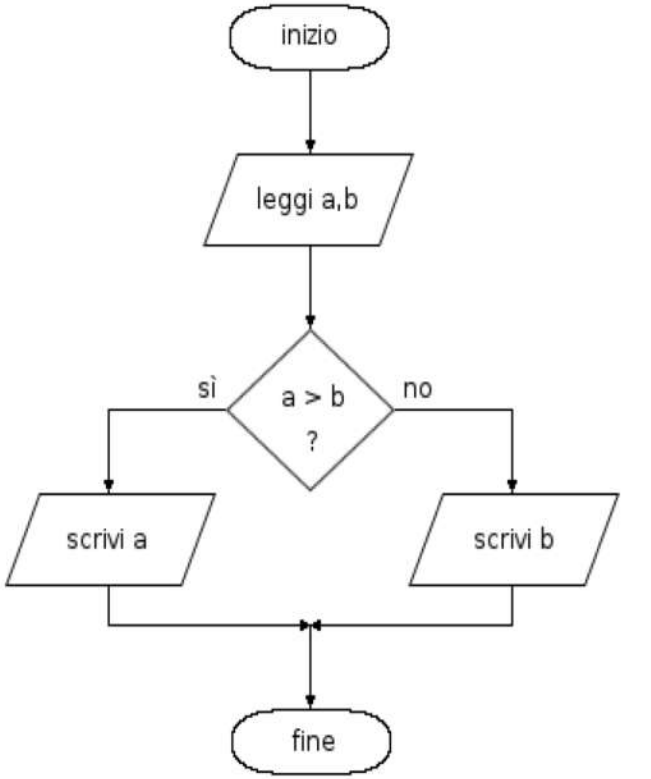


Diagramma a blocchi

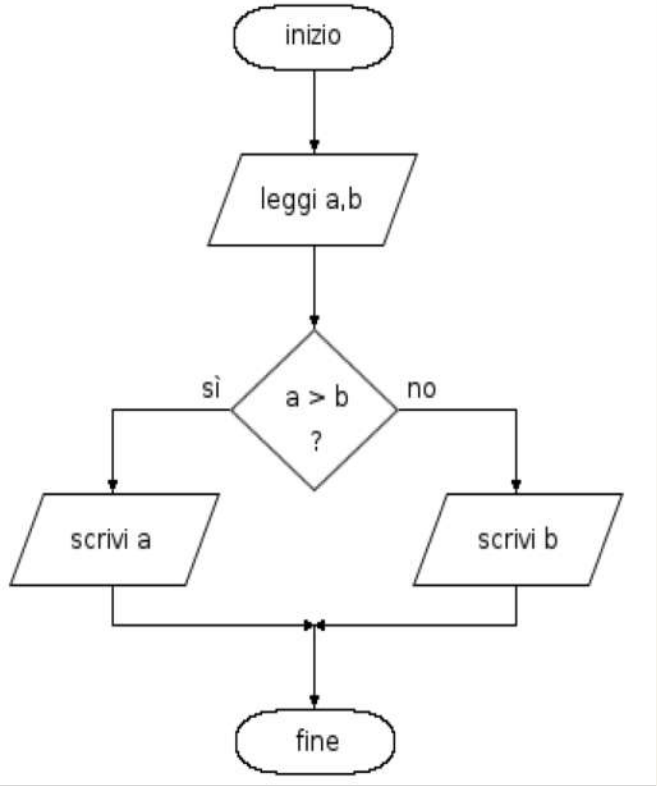
Diagrammi a blocchi

- Un particolare simbolo grafico detto blocco elementare è associato a ciascun tipo di istruzione elementare.
- I blocchi sono collegati fra loro tramite frecce che indicano il susseguirsi delle istruzioni.



Diagrammi a blocchi

- È un linguaggio formale di tipo grafico per rappresentare gli algoritmi.
- Attraverso il diagramma a blocchi (o flow chart) si può indicare l'ordine di esecuzione delle istruzioni.



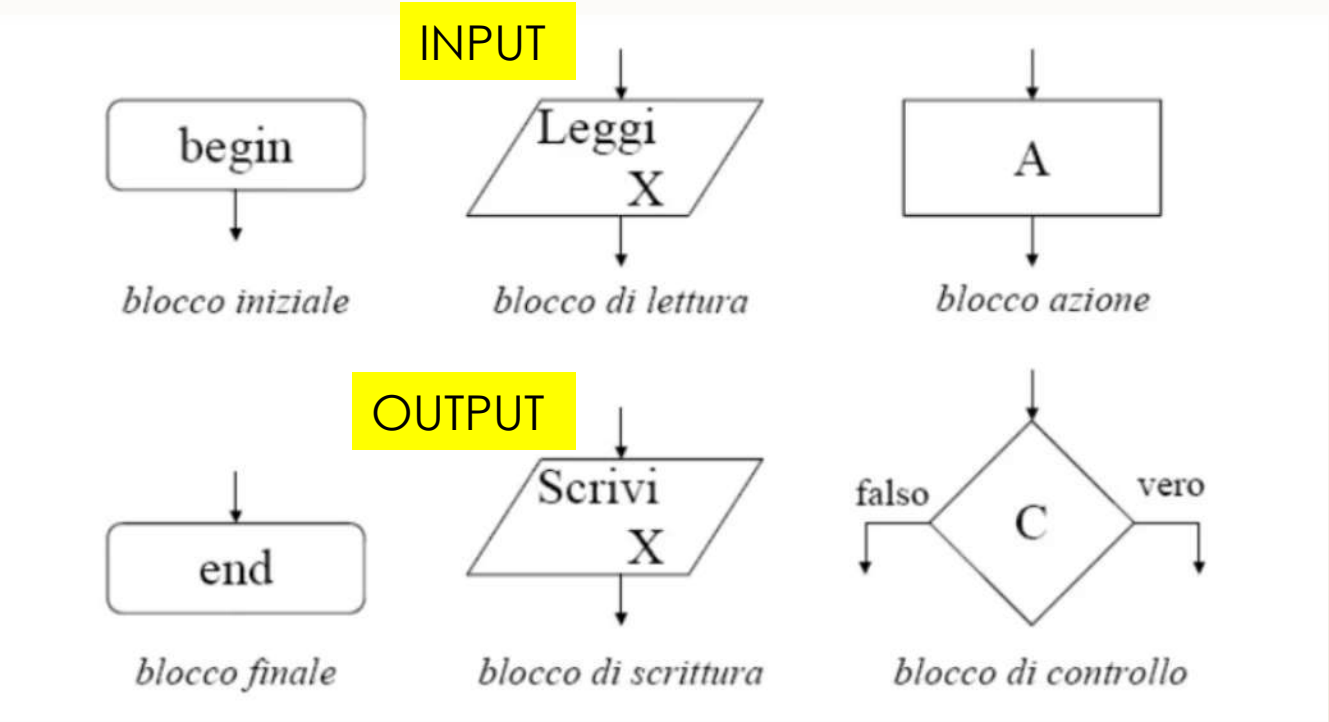
Diagrammi a blocchi

- Ogni diagramma inizia con:

- E termina con

Diagrammi a blocchi

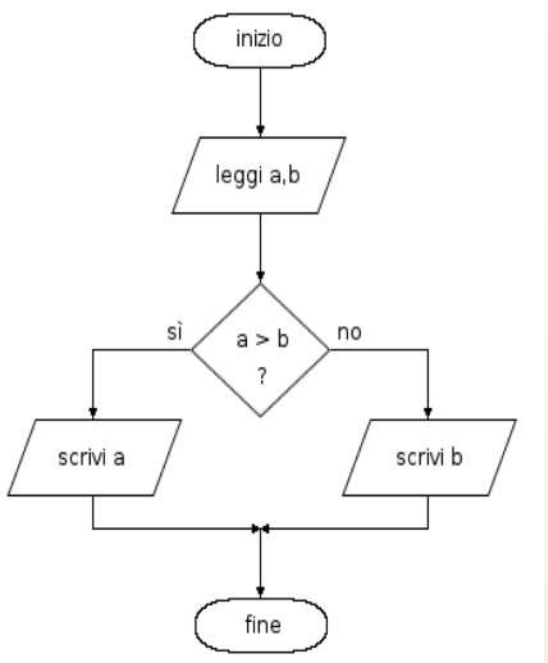
- I blocchi elementari sono i seguenti



Diagrammi a blocchi

Un diagramma a blocchi descrive un algoritmo se:

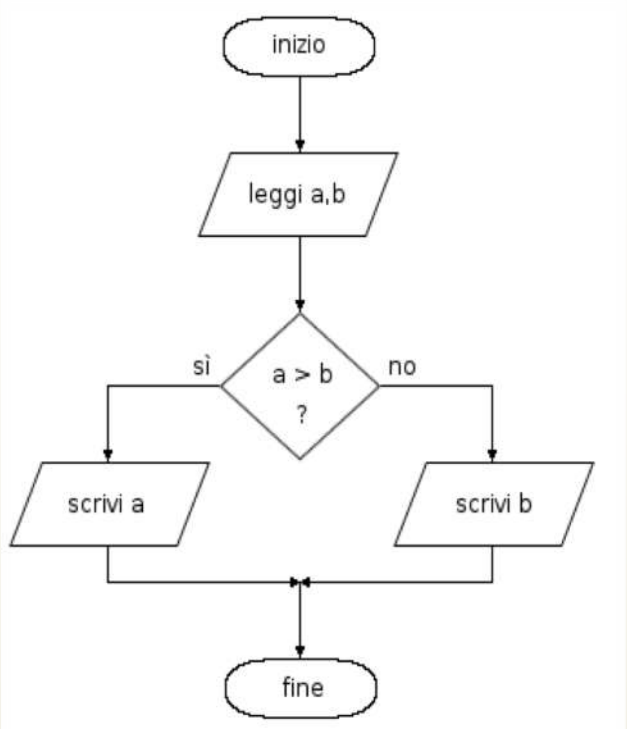
- ha un blocco iniziale e uno finale;
- è costituito da un numero finito di blocchi azione e/o blocchi lettura/scrittura e/o blocchi di controllo;
- ciascun blocco elementare soddisfa le condizioni di validità.



Diagrammi a blocchi

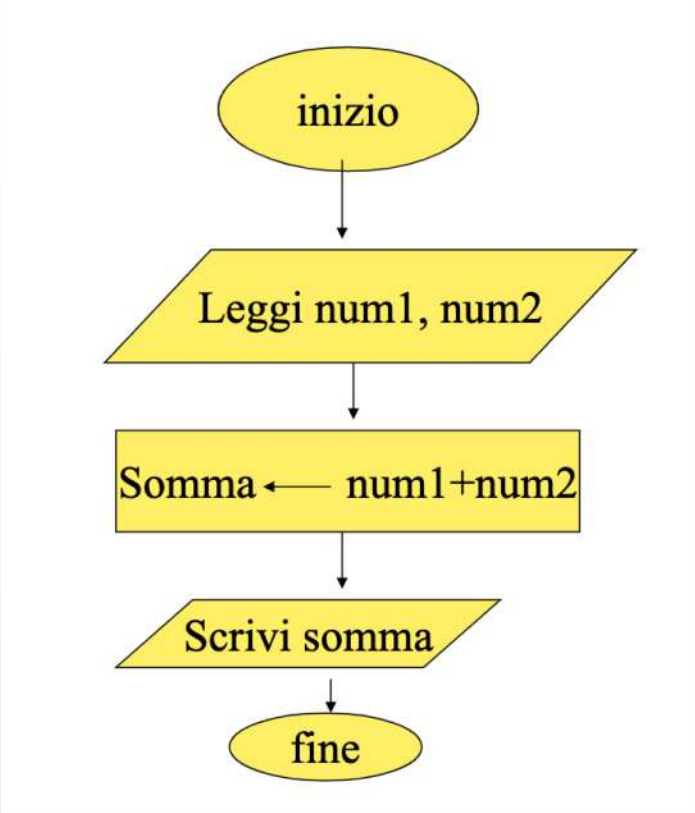
Condizioni di validità:

- ciascun blocco azione, lettura/scrittura ha una sola freccia entrante e una sola freccia uscente;
- ciascun blocco di controllo ha una sola freccia entrante e due frecce uscenti;



Esempio

Es. di diagramma di flusso per il calcolo della somma di due numeri



Esempio di pseudocodifica

PROGRAMMA somma

INIZIO

LEGGI (num1, num2)

somma num1+num2

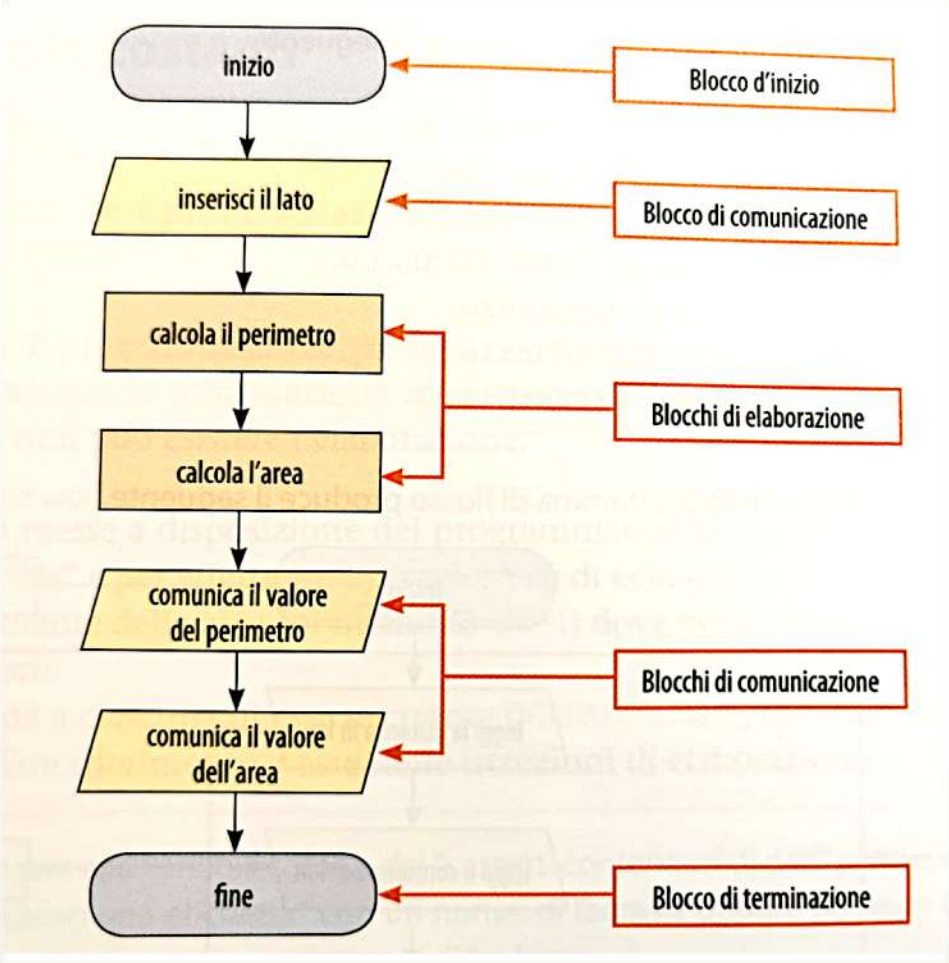
SCRIVI (somma)

FINE

Esercitazione

1. Descrivere l'algoritmo che elenca le operazioni necessarie per calcolare il perimetro e l'area di una di un quadrato letto la misura del lato.
2. Descrivere l'algoritmo che calcoli la percentuale di sconto del valore di un determinato bene.
3. Date la temperatura di tre stanze differenti calcolare la temperatura media.
4. Dato il numero di abitanti di un Comune di due anni fa e la popolazione attuale. Calcolare se il comune ha avuto un aumento o una diminuzione della popolazione e la rispettiva percentuale.
5. Descrivere l'algoritmo che calcoli il valore scontato se il prezzo è maggiore di 50 si applica il 5% di sconto, altrimenti il 2%.

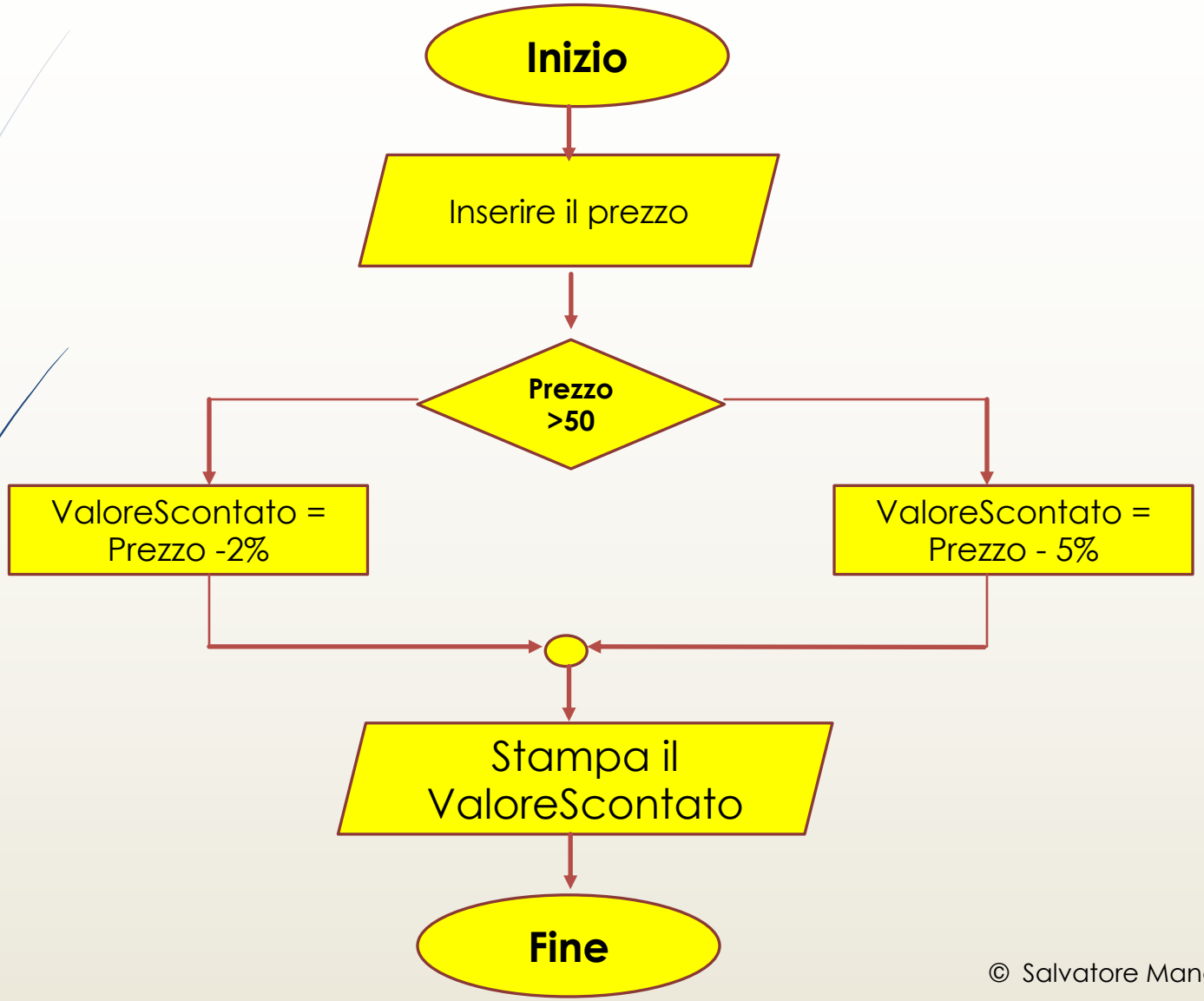
Soluzione Es. n°1



Analisi del problema

- Descrivere l'algoritmo che calcoli il valore scontato se il prezzo è maggiore di 50 si applica il 5% di sconto, altrimenti il 2%.
  - Input → Prezzo
  - Output → Valore Scontato
  - Calcolo → Lo sconto da applicare

Soluzione Es. n°3

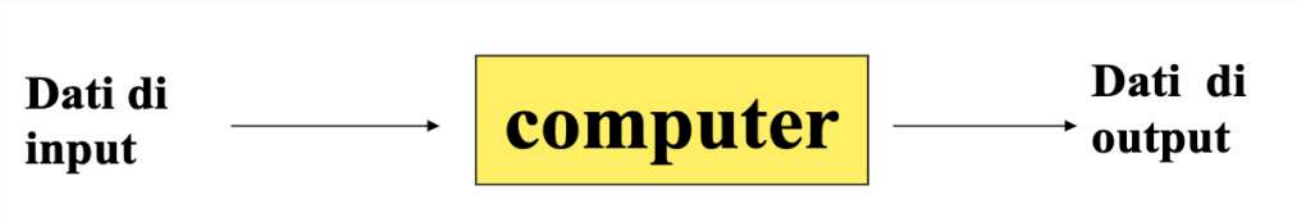


Esercitazione

Devi organizzare un viaggio con i tuoi amici, devi esaminare i costi, confrontando le spese del viaggio in autobus o in treno per raggiungere la meta. Sulla base dei calcoli effettuati, devi decidere qual è il mezzo di trasporto più conveniente

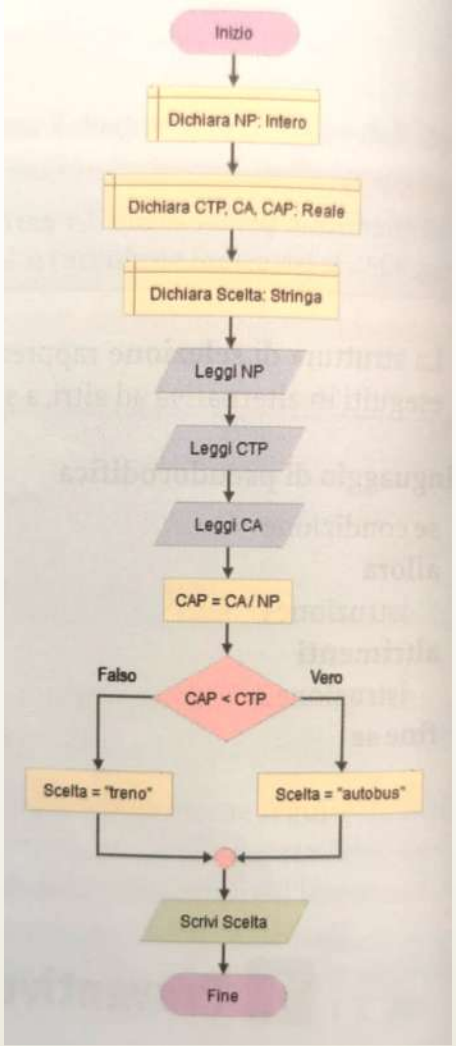
Analisi del problema

- **Dati di input**
  - Numero dei partecipanti → **NP**
  - Costo del treno per persona(andata e ritorno) → **CTP**
  - Costo complessivo dell'autobus → **CA**
- **Dati di output**
  - Scelta più conveniente tra treno e autobus



Soluzione del problema

- **Dati di input**
  1. Acquisire il numero dei partecipanti → **NP**
  2. Acquisire il costo del treno per persona → **CTP**
  3. Il costo complessivo dell'autobus → **CA**
- **Dati di calcolo**
  1. **CAP** → Costo dell'autobus per persona = **CA/NP**
- **Dati di output**
  - **SE** il CAP è minore di CTP
    - la scelta è **Autobus**
  - **ALTRIMENTI**
    - la scelta è **Treno**



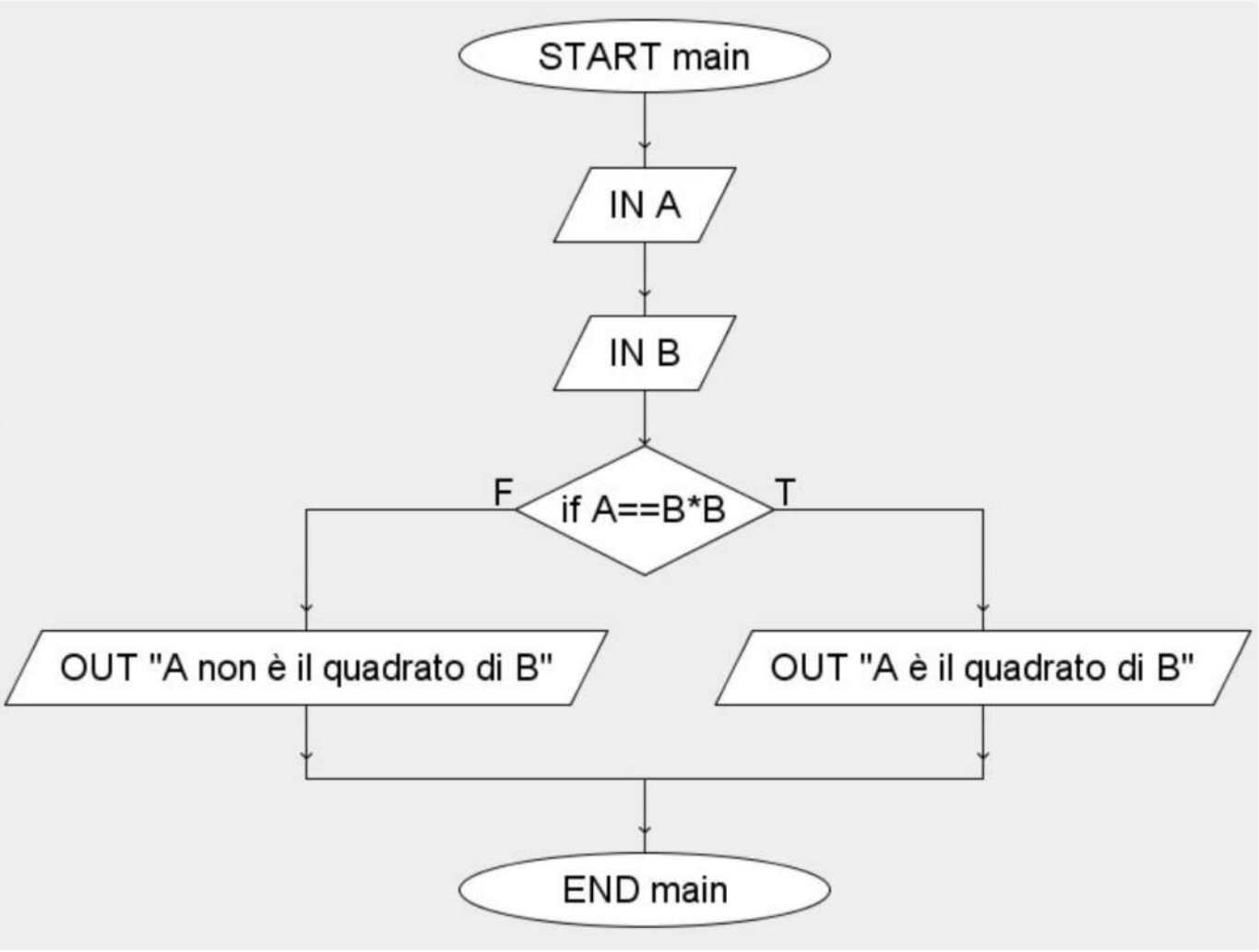
Esercitazioni

1. Dati due numeri A e B verificare se A è il quadrato di B
2. Data una temperatura **T**, visualizzare se si tratta di una temperatura: "sotto lo zero", "uguale a zero" o "sopra lo zero". Visualizzare il messaggio in output.
3. Date le dimensioni di due rettangoli calcolarne l'area e determinare quale dei due ha la superficie maggiore.
4. Realizziamo un algoritmo per il calcolo del valore **massimo fra tre numeri** presi in input, ovvero il numero maggiore.

Soluzione 1

Dati due numeri A e B verificare se A è il quadrato di B

- Prendiamo in input A e B utilizzando il parallelogramma.
- Dopo utilizzare il rombo per verificare se A è uguale al quadrato di B. Cioè poniamo come test: A==B*B.
- Se la condizione è vera visualizziamo semplicemente in output, utilizzando il parallelogramma, il messaggio '*A è il quadrato di B*'.
- Altrimenti se la condizione è falsa visualizziamo in output il messaggio: '*A non è il quadrato di B*'.

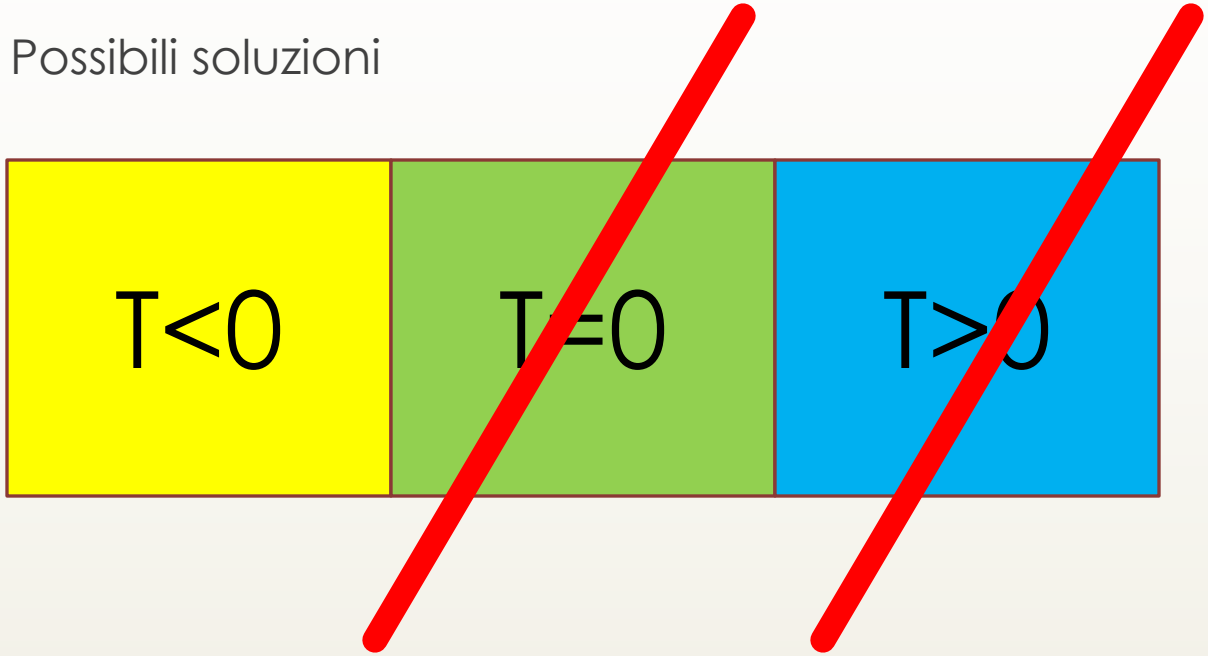


Soluzione n°2

Data una temperatura **T**, visualizzare se si tratta di una temperatura: **“sotto lo zero”**, “**uguale a zero**” o “**sopra lo zero**”. Visualizzare il messaggio in output.

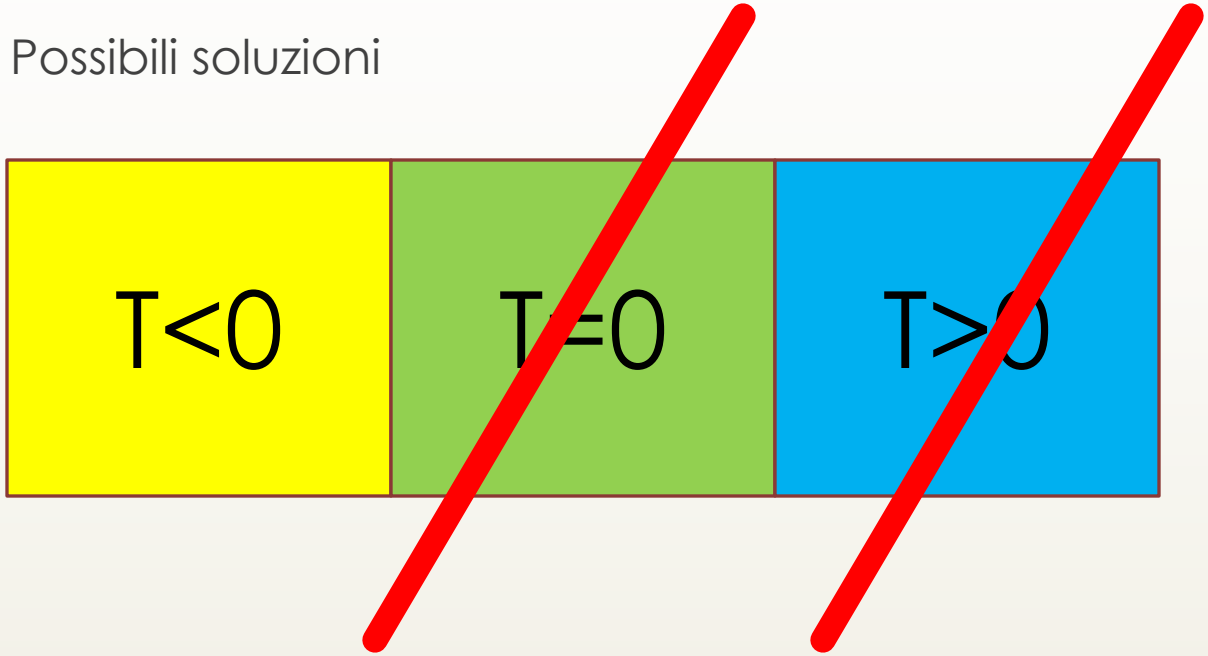
Soluzione n°2

- Possibili soluzioni

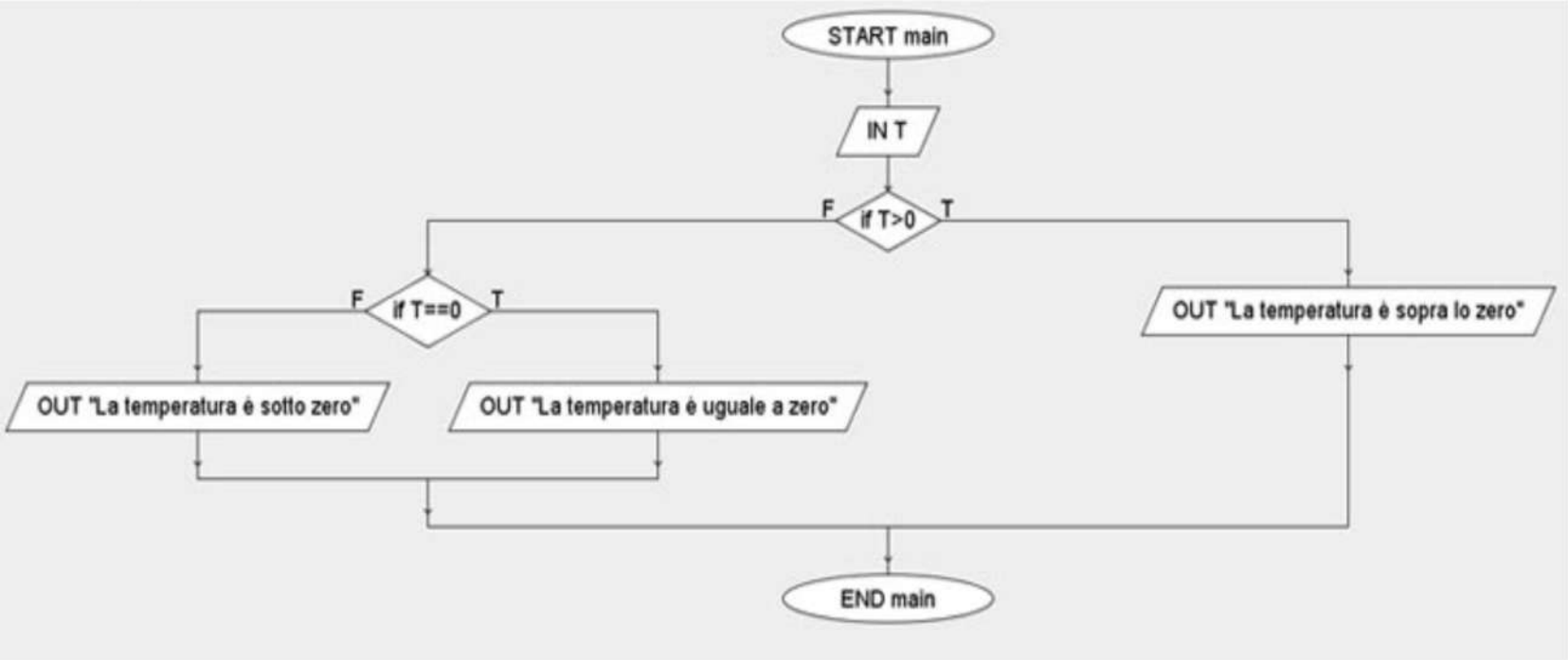


Soluzione n°2

- Possibili soluzioni



Soluzione n°2



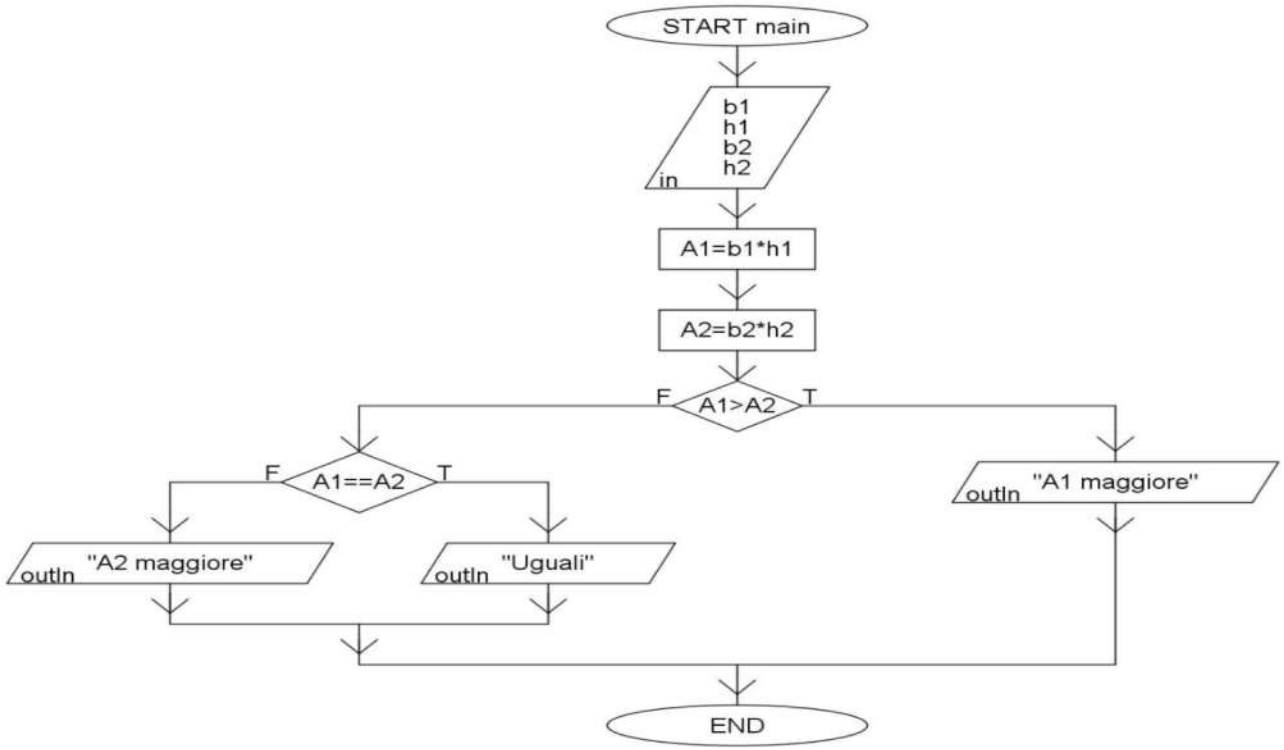
Soluzione n°3

- Date le dimensioni di due rettangoli calcolarne l'area e determinare quale dei due ha la superficie maggiore.
- Per risolvere questo algoritmo dobbiamo prendere in input i dati necessari per calcolare l'area dei due rettangoli.
- Quindi prendiamo in input:
- **b1** – indica la base del primo rettangolo
- **h1** – indica l'altezza del primo rettangolo
- **b2** – indica la base del secondo rettangolo
- **h2** – indica l'altezza del secondo rettangolo
- Dopo calcoliamo l'area dei due rettangoli utilizzando due variabili A1 e A2.

Soluzione n°3

- Infine confrontiamo le due aree così ottenute per vedere quale delle due è maggiore. Quindi se A1 è maggiore di A2 scriviamo che A1 è maggiore. Altrimenti non possiamo ancora dire che A2 è maggiore di A1 in quanto dobbiamo verificare se sono uguali.
- Ecco quindi il diagramma di flusso che rappresenta l'algoritmo proposto.
- Notate che abbiamo messo le 4 variabili b1, h1, b2 e h2 per comodità nello stesso input.
-

Soluzione n°3

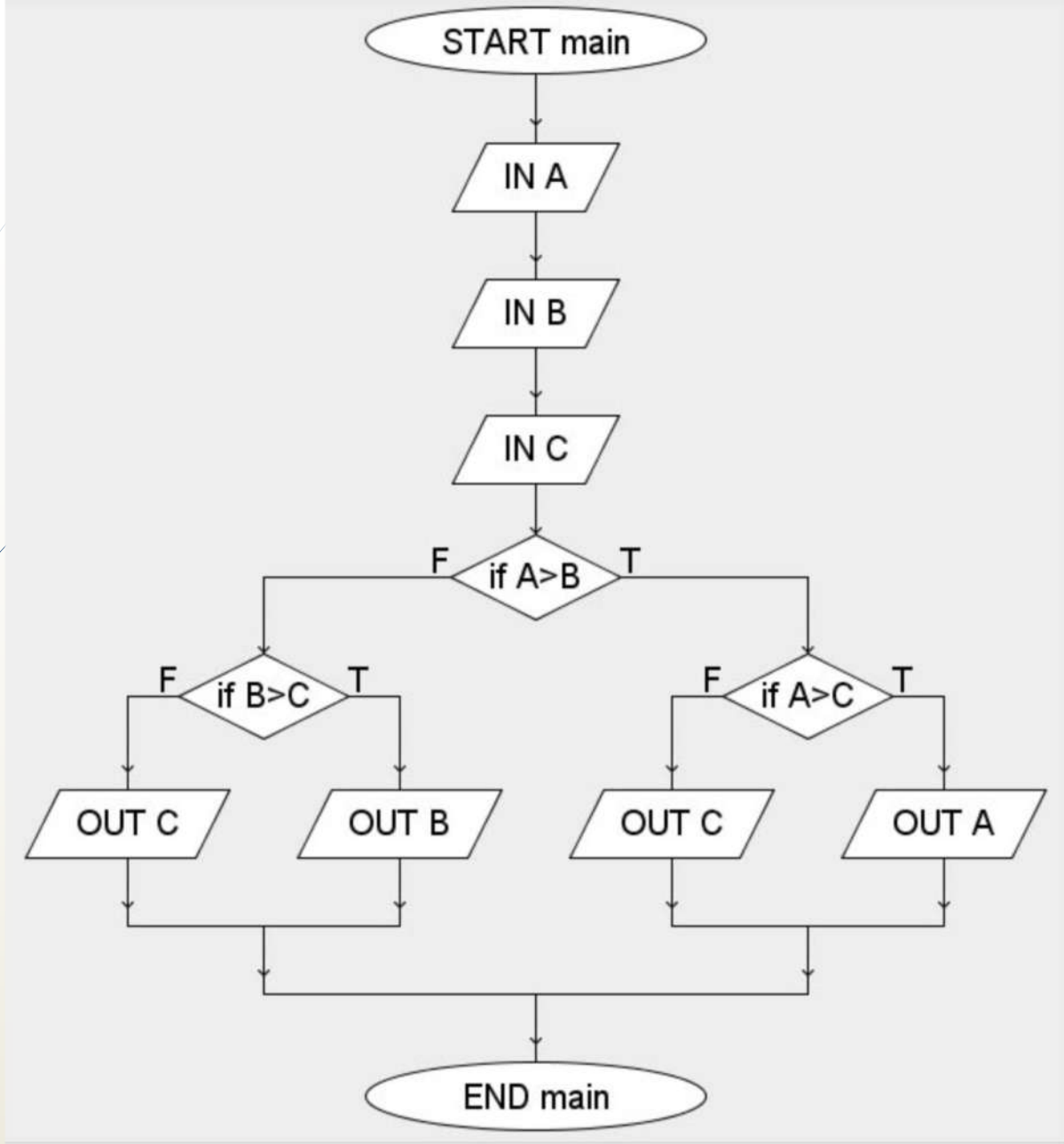


Soluzone n°4

Realizziamo un algoritmo per il calcolo del valore **massimo fra tre numeri** presi in input, ovvero il numero maggiore.

Soluzione n°4

- Innanzitutto prendiamo in input i tre numeri: A, B e C.
- Dopo effettuiamo il test **A>B** cioè ci chiediamo **A è maggiore di B**?
- Abbiamo allora due possibilità:
  1. Se il test è **vero** possiamo tralasciare B e confrontare A con C. Ci servirà dunque un altro rombo per effettuare il secondo test
  2. Se il test iniziale (**A>B**) è **falso** allora vuol dire che A è minore di B, quindi sicuramente A non sarà il maggiore, pertanto confronto B con C, in questo modo: **B>C**



Gli operatori Logici

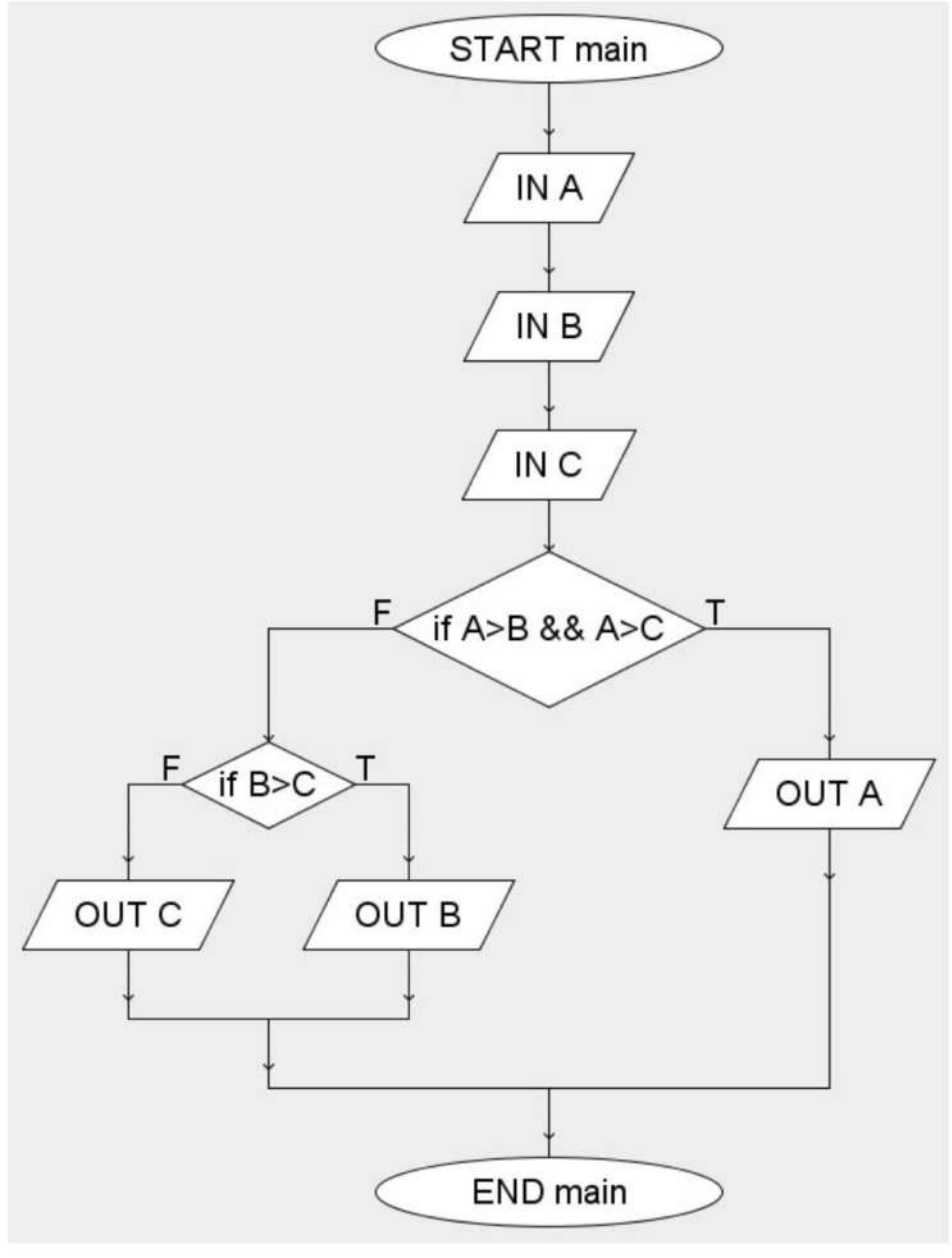
- In questa soluzione utilizziamo l'operatore || che sta per **or**, cioè la funzione logica **O**.
- In questa soluzione utilizziamo l'operatore **&&** che sta per **and**, cioè la funzione logica **E**.

Soluzione con gli operatori logici

- L'algoritmo si può risolvere anche in un altro modo. Ad esempio utilizzando gli operatori logici.
- In questa soluzione utilizziamo l'operatore **&&** che sta per **and**, cioè la funzione logica **E**.
- Poniamo come prima condizione che A sia maggiore di B e contemporaneamente che A sia maggiore di C. Dunque in questo modo: A>B && A>C.

Soluzione con gli operatori logici

- Se tale condizione è vera chiaramente A è il maggiore.
- Altrimenti se A>B && A>C è falsa, vuol dire che A non può essere il maggiore, perché chiaramente non è comunque maggiore di entrambi. Dunque possiamo escludere A dal prossimo confronto e valutare solo B e C.
- Quindi controlliamo solo se B è maggiore di C e se tale condizione è vera il massimo sarà B, altrimenti il massimo sarà C.



Soluzione con una variabile di comodo

- Utilizziamo un variabile di comodo **MAX** dove memorizziamo il primo valore preso in input A.
- se **B è maggiore di MAX** sostituisco il valore, altrimenti non faccio nulla in quanto MAX è più grande.
- Dopo controllo se **C è maggiore di MAX** e se vero sostituisco il valore, altrimenti come prima vuol dire che MAX è il maggiore.