

Algoritmi e Strutture Dati

Lezione 1 | 23 Febbraio

Università degli Studi di Modena e Reggio Emilia — UnimoreE

1. Cos'è un problema (algoritmico)?

Un problema non è necessariamente qualcosa di negativo. In senso algoritmico, un problema è semplicemente la descrizione di:

- **Cosa ho in partenza** (input)
- **Cosa voglio ottenere** (output)

1.1. Problemi di tipo generale

Non ci interessano problemi specifici con valori fissi. Ci interessano problemi **generali**, validi per qualsiasi configurazione di input valida.

- × Una calcolatrice che sa fare solo $3 + 5$
- ✓ Una calcolatrice che sa fare la somma di **qualsiasi** coppia di interi

Per questo **specifichiamo il tipo dei valori**, non i valori stessi.

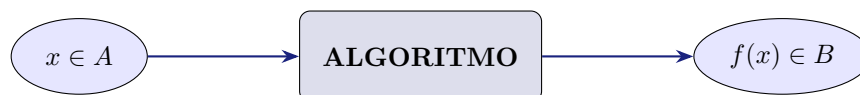
Input	Output
Due interi x e y	La somma $x + y$
Una sequenza di n numeri	Il massimo della sequenza

2. Il problema come funzione

Possiamo vedere un problema come una **funzione matematica** che associa ogni configurazione di input al corrispondente output:

$$f : A \longrightarrow B$$

$$x \in A \xrightarrow{\text{algoritmo}} f(x) \in B$$



Nota

Il problema *mappa* l'input all'output. Non è detto che la funzione sia biunivoca: più input diversi possono dare lo stesso output.

3. Istanza di input

Un problema è definito in modo generale, ma per utilizzarlo servono valori concreti.

Definizione

Istanza di input = una specifica configurazione di input in cui i valori rispettano i vincoli imposti dal problema.

Attenzione

Il numero di istanze possibili è (in genere) **infinito**.

3.1. Esempio

Problema: “Due interi x e y tali che $x < y$ ”

Istanza	Valida?	Motivo
$x = 2, y = 5$	✓ Sì	$2 < 5$
$x = 7, y = 3$	× No	$7 \not< 3$

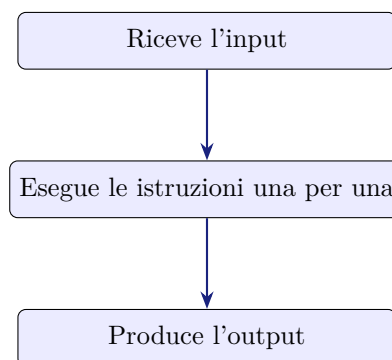
4. Cos'è un algoritmo?

Dire “dato questo input, voglio quest'output” è solo l'inizio del lavoro. Il **come** ci si arriva è il compito dell'algoritmo.

Un algoritmo è la sequenza di istruzioni che, se seguita, permette di passare dall'input all'output — **anche senza capire il perché**.

4.1. L'esecutore

Serve qualcuno (o qualcosa) che esegua l'algoritmo: l'**esecutore**.

**Nota**

L'esecutore può non avere alcuna idea del risultato finale — eppure, seguendo le istruzioni, ci arriverà. Gli esecutori non sono necessariamente computer: nella vita scolastica siamo stati esecutori di moltissimi algoritmi (es. le procedure di calcolo dell'aritmetica).

I linguaggi di programmazione stabiliscono quali istruzioni un computer è in grado di eseguire — non possiamo inventarcele.

5. Definizione formale di algoritmo

Algoritmo

Sequenza finita di istruzioni **ben ordinate**, **non ambigue** ed **effettivamente realizzabili** che, eseguita, produce un risultato e **termina** in una quantità finita di tempo.

5.1. Requisiti uno per uno

Requisito	Significato
Sequenza finita	Non può essere una lista infinita di istruzioni
Ben ordinata	L'ordine è sempre noto; dopo ogni istruzione non vi è ambiguità su quale sia la successiva
Non ambigua	Ogni istruzione deve essere interpretabile in un solo modo dall'esecutore
Effettivamente realizzabile	L'esecutore deve essere in grado di eseguire ogni istruzione (es. non chiedere di dividere per zero)
Termina	Deve fermarsi in un tempo finito
Produce un risultato	Deve restituire un output (anche solo un messaggio di errore, purché deterministico)

6. Alan Turing e la macchina astratta

La prima definizione matematica di algoritmo è stata formulata da **Alan Turing**.

Turing non si è limitato a decifrare i codici tedeschi: ha studiato **macchine astratte** che risolvono problemi. La sua intuizione fondamentale:

Intuizione di Turing

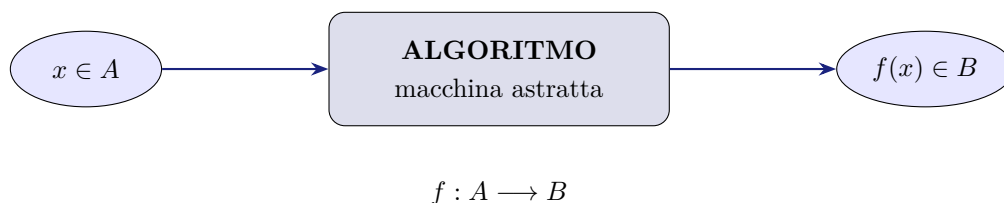
Un algoritmo è una **macchina astratta** che trasforma una sequenza arbitraria x di simboli (il dato) in una sequenza y di simboli (il risultato).

Questo non è altro che calcolare una funzione:

$$y = f(x)$$

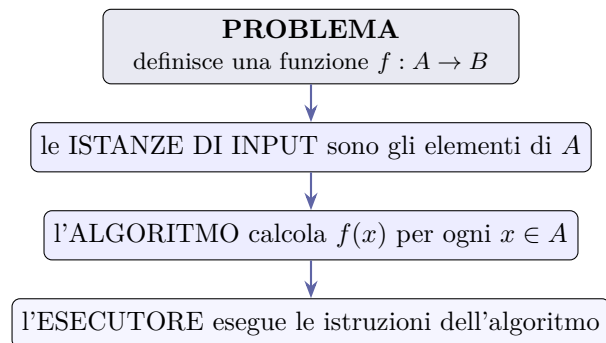
con cui risolvere il problema corrispondente.

6.1. Schema della macchina astratta



7. Riepilogo concettuale

I concetti introdotti sono strettamente collegati in una gerarchia:



Concetto	Descrizione sintetica
Problema	Descrizione astratta di input e output
Istanza di input	Una specifica configurazione concreta di input valida
Algoritmo	Procedura finita, non ambigua, che calcola $f(x)$ e termina
Esecutore	Entità (umana o macchina) che segue le istruzioni