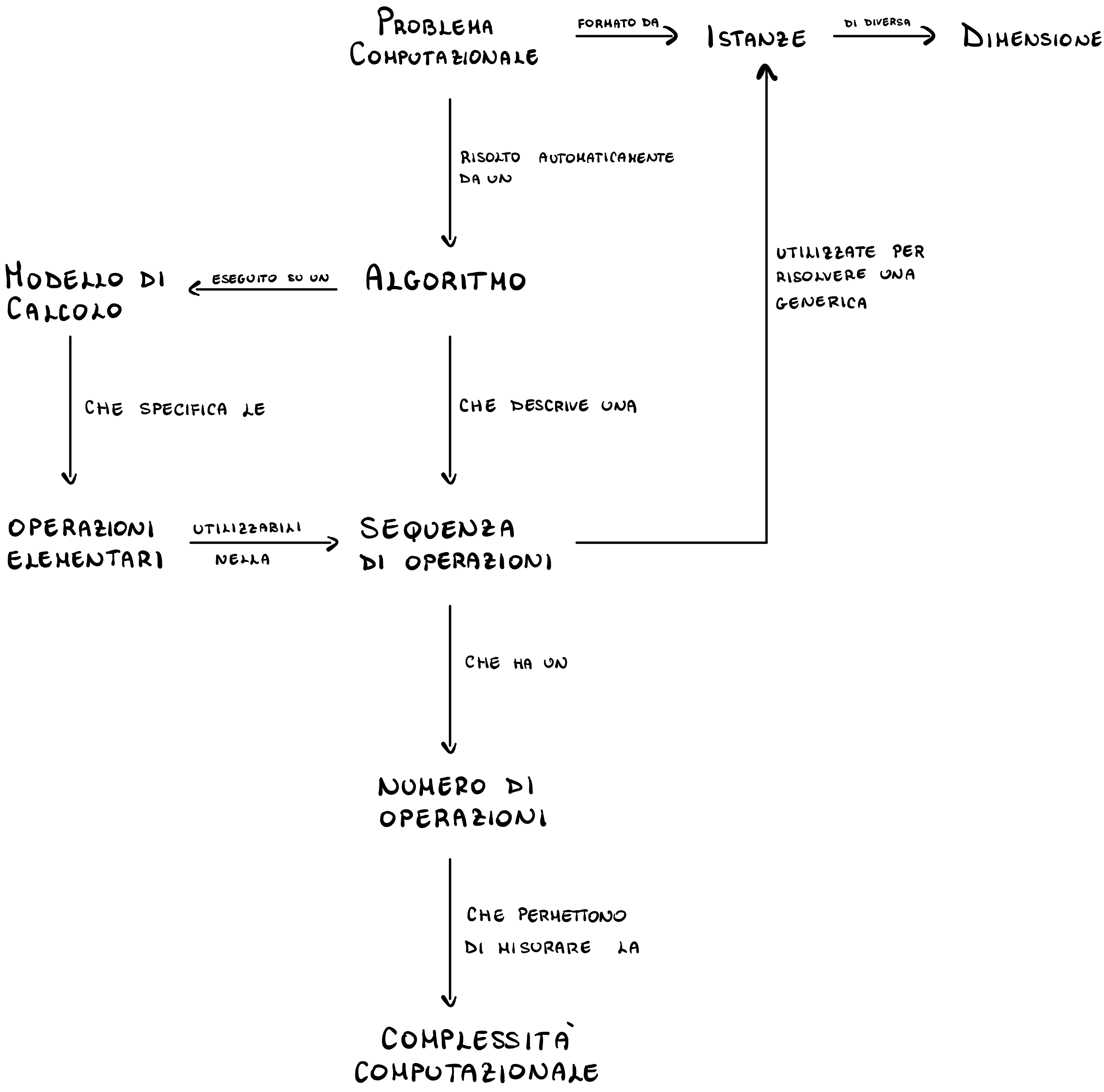
Schema Generale



Problema Computazionale

PROBLEMI che possono potenzialmente essere risolti tramite una COMPUTAZIONE, ovvero tramite un PROCESSO AUTOMATICO che non necessita alcun tipo di <u>CREATIVITÀ</u> o di <u>INTUIZIONE</u> umana.

ORDINAMENTO (SORTING)

Data una SEQUENZA di numeri, RITORNARE la sequenza ORDINATA

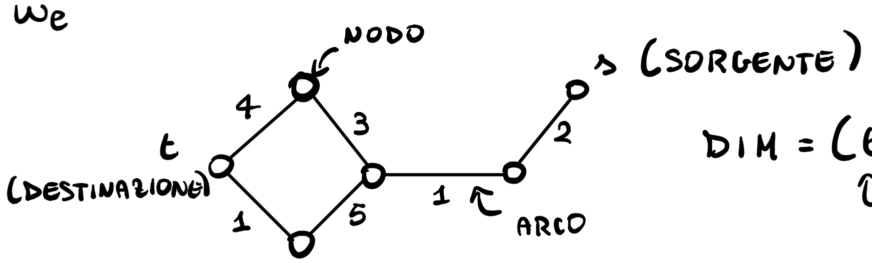
(INPUT)		(OUTPUT)	
• [2, 3, 1, 0]	→	[0, 1, 2, 3]	, DIM = 4
• [10, 0, 2]	→	[0, 2, 10]	, DIM = 3
• [5, -5, 1]	→	[-5, 1, 5]	, DIM = 3

$[a_1, a_2, \dots, a_n] \rightarrow [a_{i_1}, a_{i_2}, \dots, a_{i_n}]$ t.c. $a_{i_1} \leq a_{i_2} \leq \dots \leq a_{i_n}$

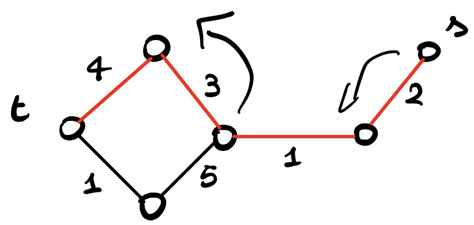
DIM = n

CAMMINO MINIMO (SHORTEST PATH)

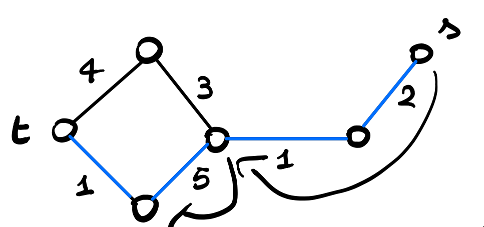
Dato un GRAFO $G = (V, E, w)$ e due NODI $s, t \in V$, trovare un CAMMINO $\pi_{s,v}$ che collega s a v MINIMIZZANDO il peso totale $\sum_{e \in \pi_{s,v}} w_e$



DIM = (6, 6)
# di archi
# nodi



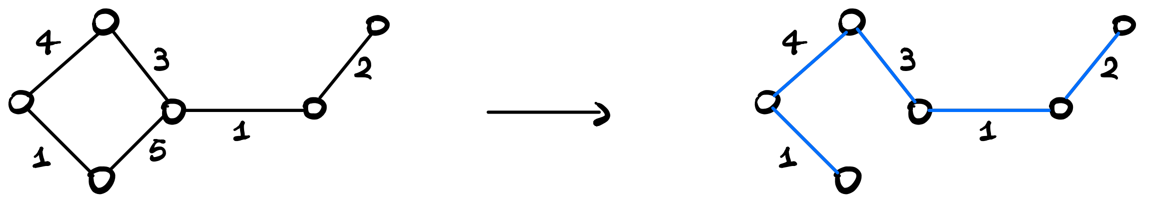
PESO TOTALE = 10



PESO TOTALE = 9 ✓

Minimo Albero Ricoprente (Minimum Spanning Tree)

Dato un grafo $G=(V,E,w)$, trovare un albero ricoprente $T=(V,E')$, $E' \subseteq E$ che minimizza il peso totale $\sum_{e \in T} w_e$



INPUT

PESO TOTALE = 11

OUTPUT

Algoritmi e Modelli di Calcolo

Un algoritmo è una sequenza di operazioni elementari che permettono di risolvere una generica istanza di un problema computazionale.

(INPUT) [2, 3, 1, 0] → $\boxed{\text{ALGORITMO}}$ → (OUTPUT) [0, 1, 2, 3]

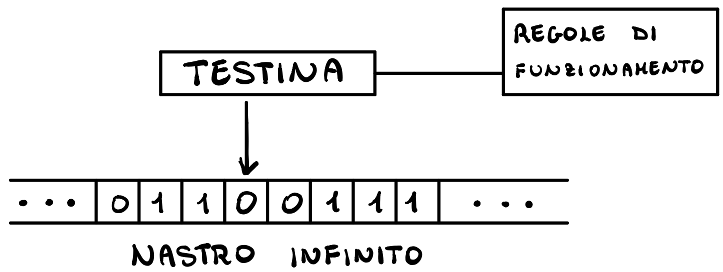
ISTRUZIONE 1
ISTRUZIONE 2
⋮
ISTRUZIONE N

Il modello di calcolo specifica quali sono le possibili operazioni elementari che un algoritmo eseguito su quel modello di calcolo può utilizzare.

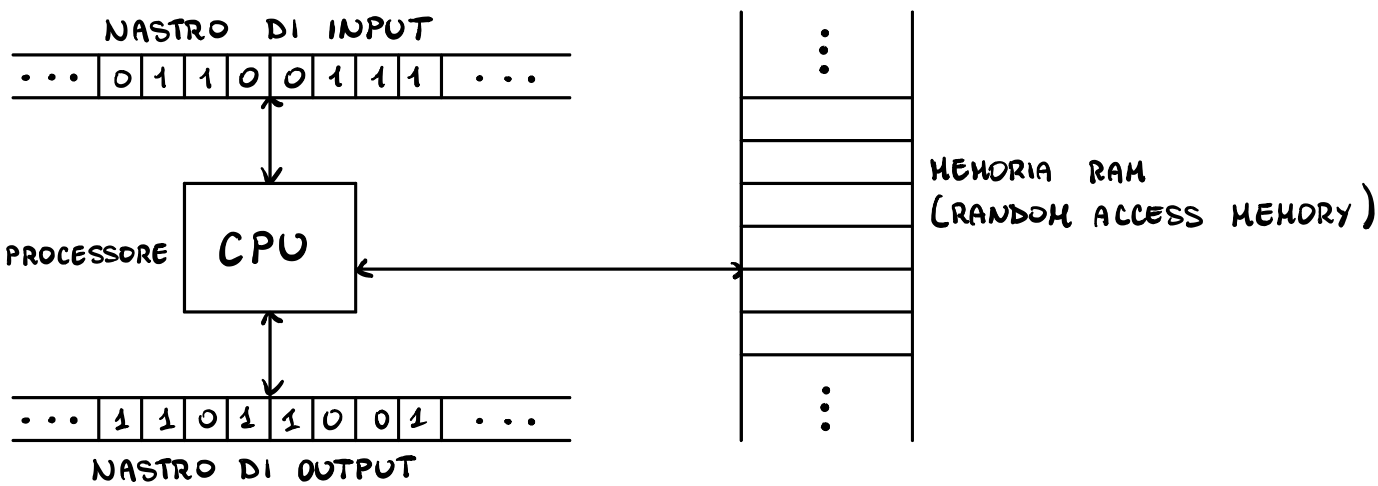
Intuitivamente possiamo effettuare la seguente ASSOCIAZIONE

CONCETTI ASTRATTI		OGGETTI REALI
ALGORITMO	⟷	SOFTWARE
MODELLO DI CALCOLO	⟷	HARDWARE

Uno dei primi modelli di calcolo fu introdotto da ALAN TURING (1912 - 1954) ed è oggi noto sotto il nome MACCHINA DI TURING.



Anche se le MACCHINE DI TURING sono molto importanti per studiare in modo formale il concetto di COMPUTAZIONE, un loro difetto è che sono troppo SEMPLICI e PRIMITIVE. Per risolvere questo problema si introduce quindi la RAM, o RANDOM ACCESS MACHINE, un modello di calcolo basato sulla famosa architettura di JOHN VON NEUMANN (1903 - 1957)



IMPORTANTE

Anche se il modello RAM presenta un livello di astrazione più elevato del modello MACCHINA DI TURING, questi due modelli sono EQUIVALENTI, ovvero tutto ciò che può essere calcolato su un modello può anche essere calcolato sull'altro modello.

ESEMPIO: (NUMERI PRIMI)

Dati due interi a, b, posso dividere a per b e ottenere altri due interi q, r tali che

$$\underbrace{a = q}_{\text{QUOZIENTE}} \cdot b + \underbrace{r}_{\text{RESTO}}, \qquad 0 \leq r < b$$

Se dividendo a con b e ottengo $r = 0$ (RESTO NULLO) allora diciamo che "b DIVIDE a".

Ad esempio,

- $4 = 2 \cdot 2 + 0$ (2 DIVIDE 4)
- $6 = 2 \cdot 3 + 0$ (3 DIVIDE 6)
- $7 = 2 \cdot 3 + 1$ (2 NON DIVIDE 7)

DEF: n è PRIMO se i suoi unici DIVISORI sono 1 e n.

Ad esempio,

10 è PRIMO? NO, 5 DIVIDE 10

7 è PRIMO? SI, nessuno tra 2, 3, 4, 5, 6 lo divide

PROBLEMA

Dato un intero $n \in \underset{\text{NUMERI NATURALI}}{\mathbb{N}} = \{0, 1, 2, \dots\}$, stabilire se n è PRIMO.

INPUT	OUTPUT
10	NO (FALSE)
7	SI (TRUE)
5	SI
20	NO

ALGORITMO 1

Per verificare se n è PRIMO scorriamo tutti i numeri tra 2 e $n-1$, dividendo ciascun numero con n. Appena troviamo un numero che divide n senza resto, ritorniamo FALSE per indicare che il numero n NON è primo. Se, dopo aver controllato tutti i numeri tra 2 e $n-1$, nessuno di questi divide n, allora ritorniamo TRUE, indicando con ciò che n è PRIMO.

Vediamo adesso lo pseudocodice

```
CHECK PRIMENESS ( integer n )
                  (INTERO)
  RESULT ← TRUE
  FOR J = 2 TO n - 1
     IF J DIVIDE n
        RESULT ← FALSE
  RETURN RESULT
```

Vediamo quindi due esempi di ESECUZIONE

- <u>$n = 7$</u>

  RESULT ← $\boxed{\text{TRUE}}$

  2 DIVIDE 7? NO

  3 DIVIDE 7? NO

  4 DIVIDE 7? NO

  5 DIVIDE 7? NO

  6 DIVIDE 7? NO

  RETURN TRUE

  ✓

- <u>$n = 35$</u>

  RESULT ← TRUE

  2 DIVIDE 35? NO

  3 DIVIDE 35? NO

  4 DIVIDE 35? NO

  5 DIVIDE 35? SI

  RESULT ← FALSE

  6 DIVIDE 35?

  7

  8

  ⋮