

# Dalla Macchina di Turing al Computer Quantistico (con un pizzico di IA)

Stefano Mangini,  
Quantum Information Theory Group **QUIT**

09/09/2021

 [stefano.mangini01@universitadipavia.it](mailto:stefano.mangini01@universitadipavia.it)

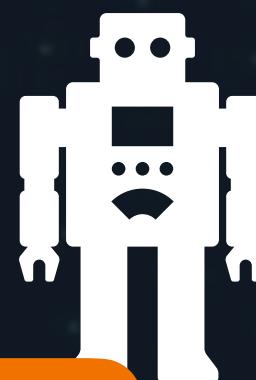
 @stfnmangini

 Stefano Mangini

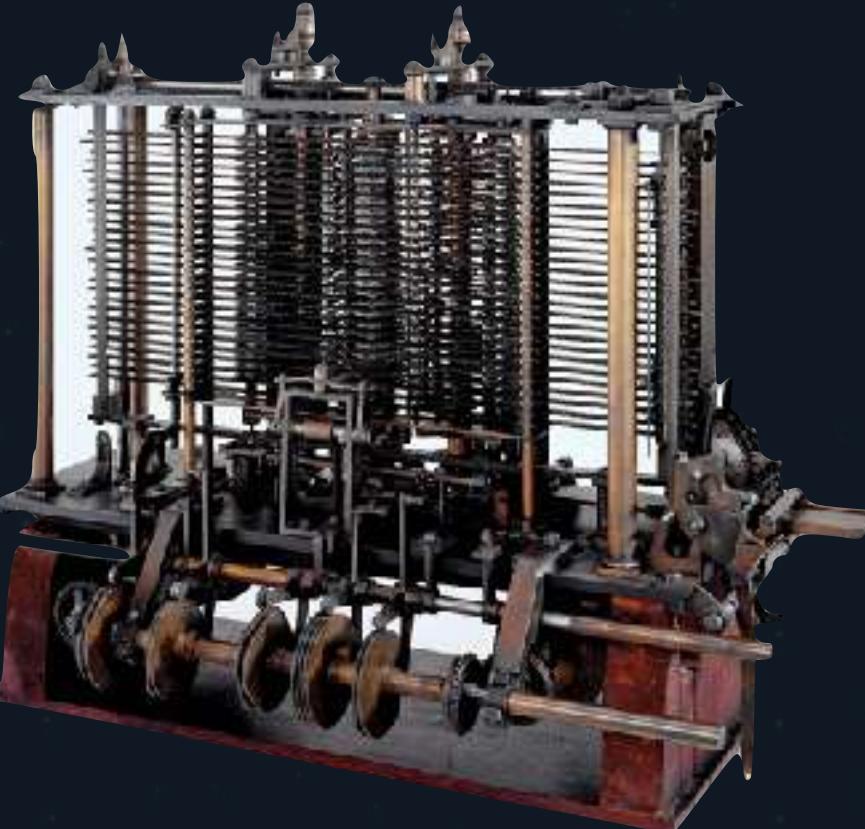
# Parte 1

# Computazione

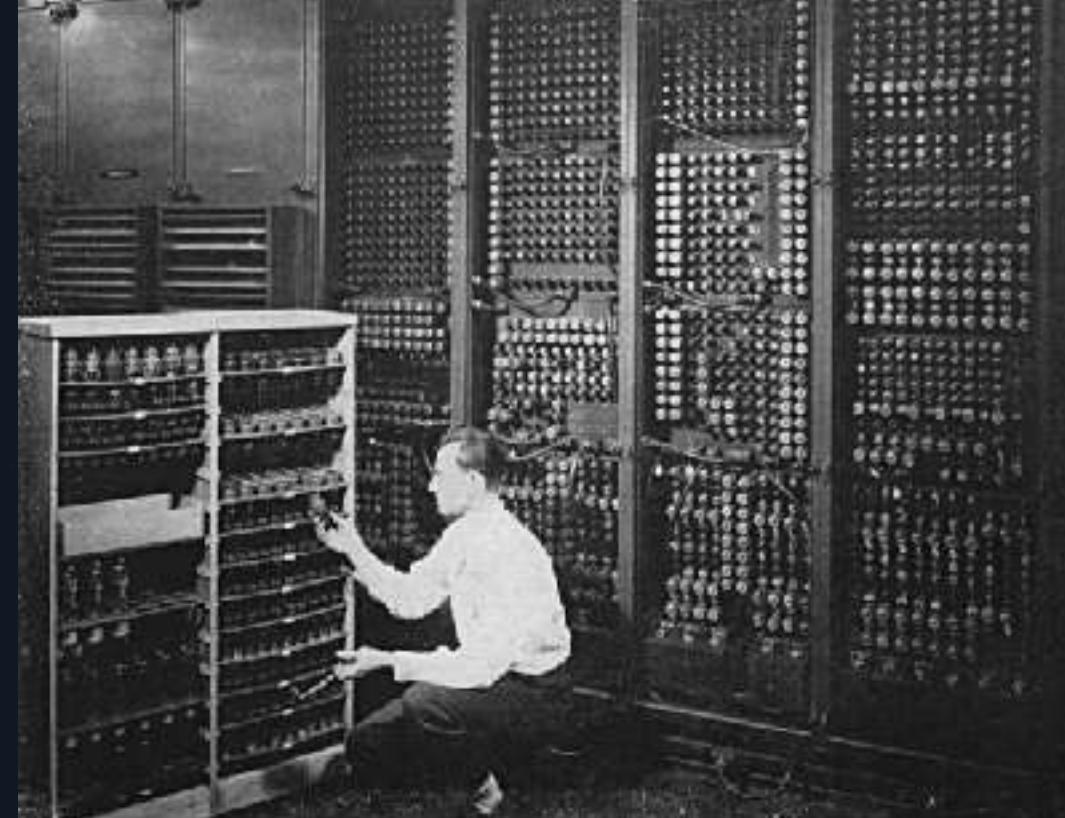
# L'evoluzione della computazione



Strumenti di calcolo



Calcolatori meccanici



Computer elettronici

Personal Computers



Laptop Cellulari



Quantum Computers

Con un pizzico di Intelligenza Artificiale

~3000AC

~1820

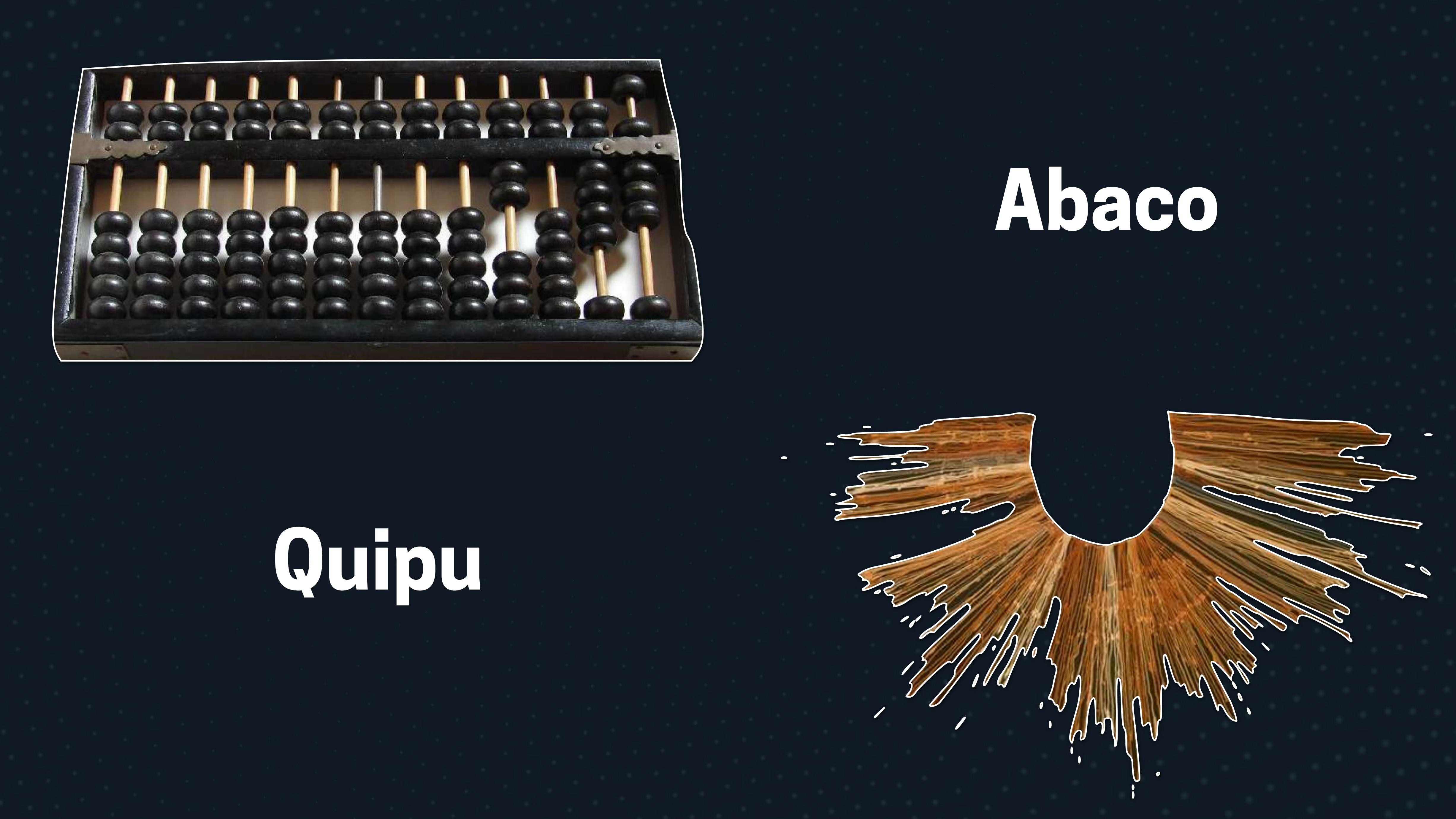
~1945

~1980

~2010

Oggi





Quipu

Abaco

510.8  
B11

Laboratory  
Lehigh University

TABLE  
OF  
LOGARITHMS  
OF THE

NATURAL NUMBERS,

FROM

1 TO 108,000.

BY CHARLES BABBAGE, ESQ.

M.A., F.R.S.L. & E., M.R.I.A., F.C.P.S.;  
MEM. ASTRON. SOC.; MEM. ACAD. DIJON; COR. PHILOMATH. SOC., PARIS;  
COR. MEMB. ACAD. MARSEILLE; AND ROYAL ACAD., BRUSSELS, ETC.

STEREOTYPED EDITION.



LONDON:  
E. & F. N. SPON, 48, CHARING CROSS.

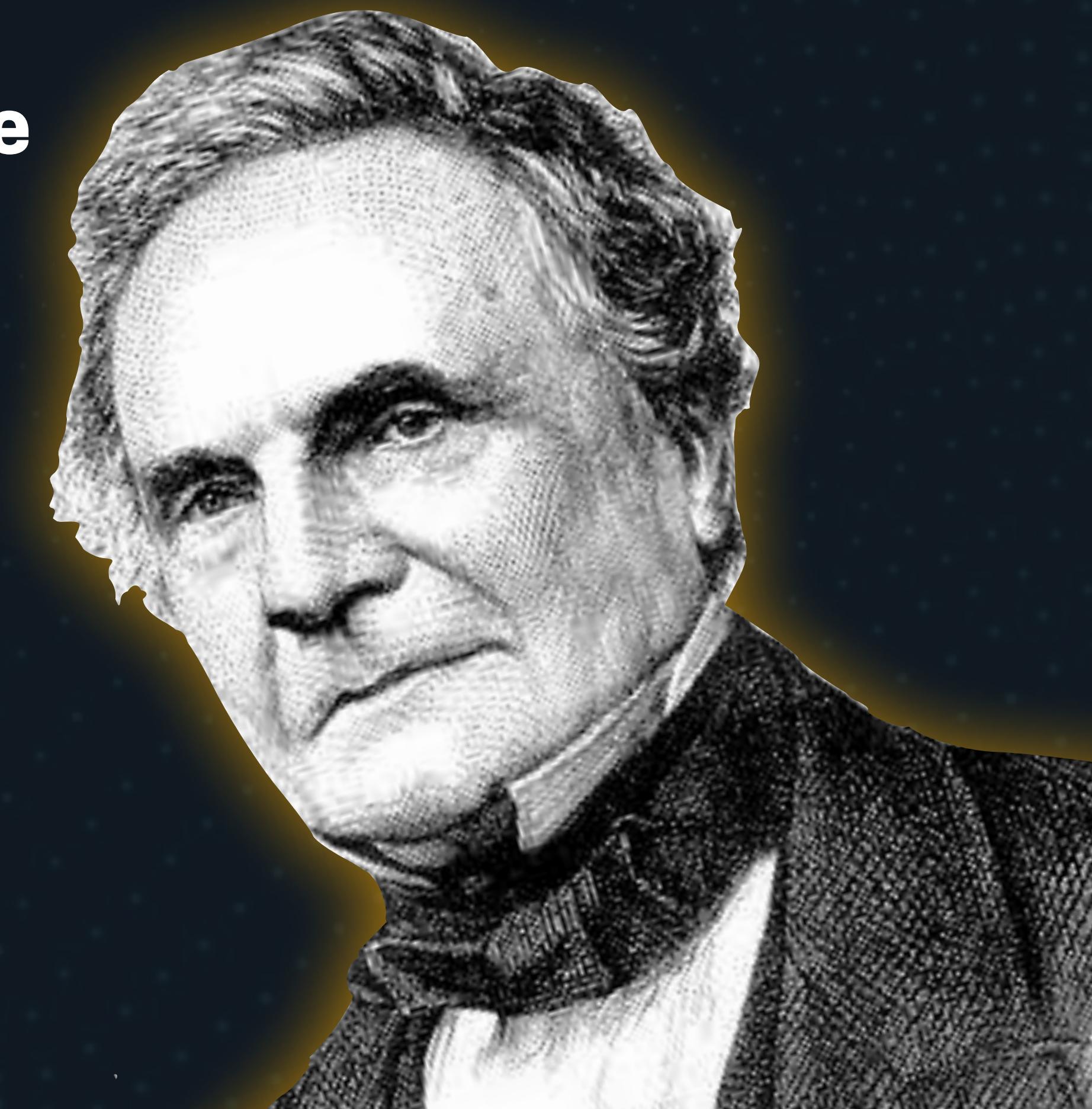
NEW YORK:  
446, BROOME STREET.

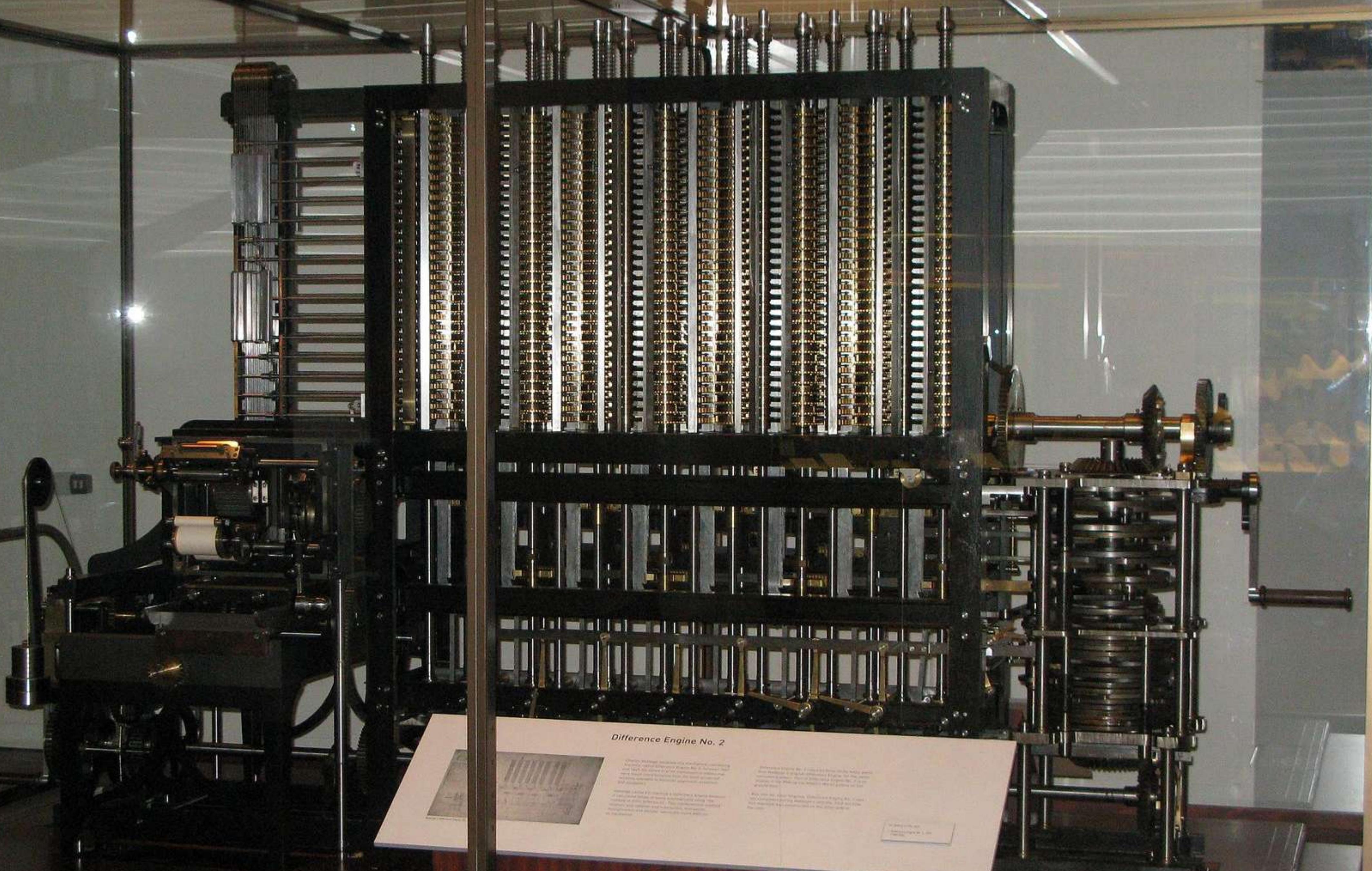
1872.

# Charles Babbage

(1791-1871)

Macchina  
differenziale





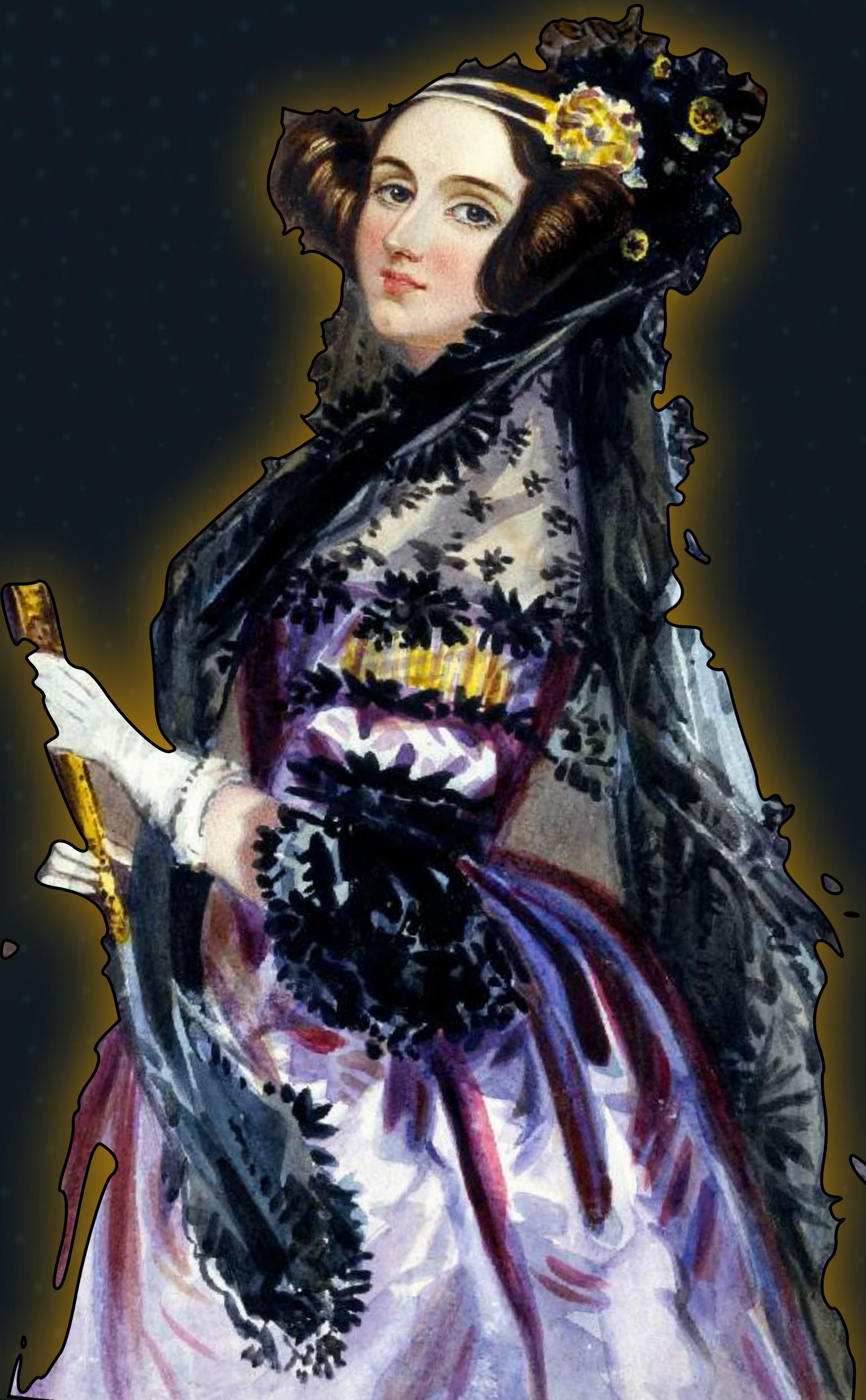
Difference Engine No. 2

Charles Babbage's reconstruction of his original Difference Engine No. 2, constructed by the Science Museum in 1991. The original engine was never completed, but this reconstruction is based on Babbage's original drawings and surviving parts.

Difference Engine No. 2 calculated tables of logarithms and trigonometric functions. It could add, subtract, multiply, divide, and calculate square roots. It also had the ability to calculate polynomials.



# Macchina Analitica



*Sketch of the Analytical Engine invented by Charles Babbage Esq. By L. F. MENABREA, of Turin, Officer of the Military Engineers.*

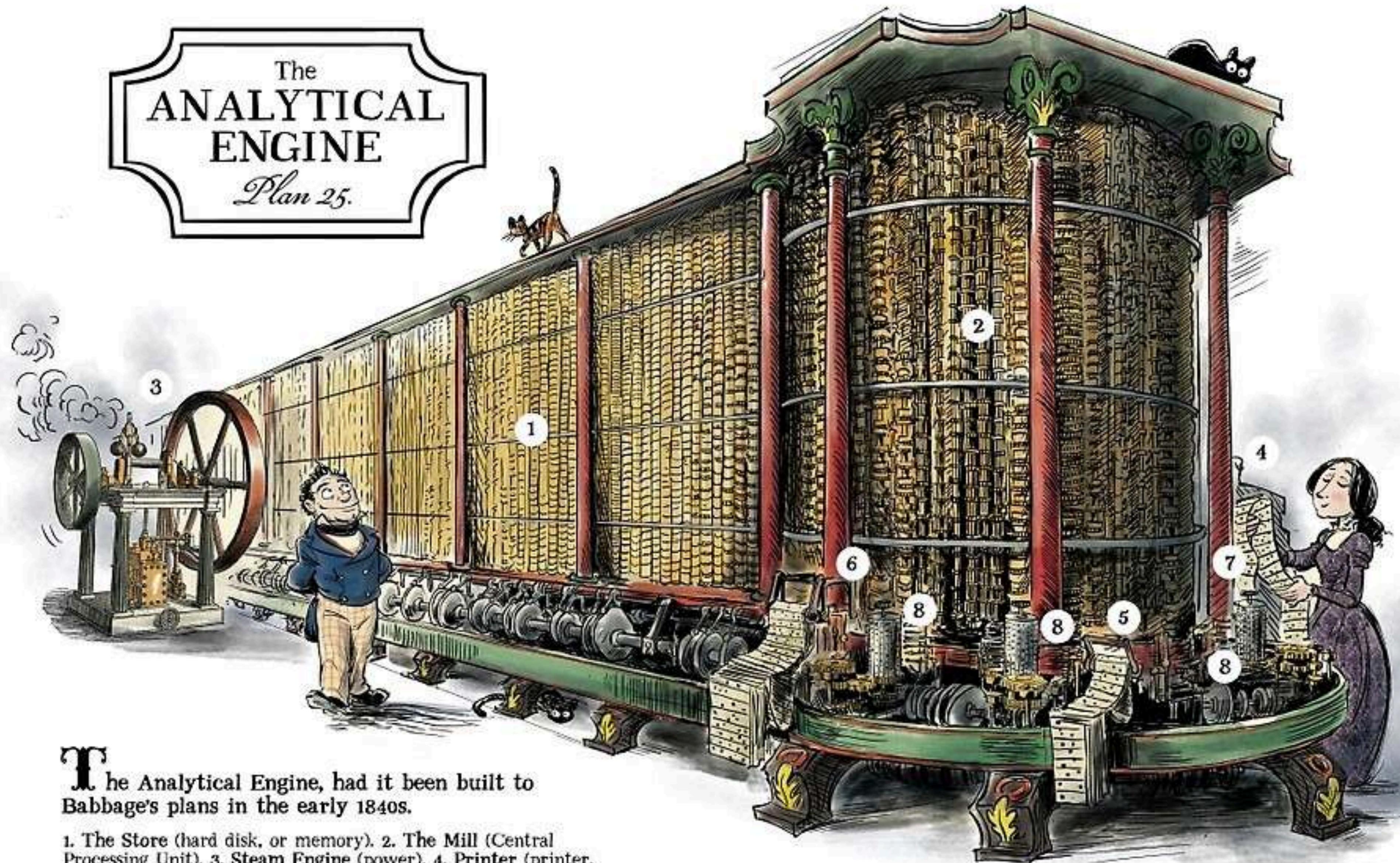
[From the *Bibliothèque Universelle de Genève*, No. 82, October 1842.]

BEFORE submitting to our readers the translation of M. Menabrea's memoir 'On the Mathematical Principles of the ANALYTICAL ENGINE' invented by Mr. Babbage, we shall present to them a list of the printed papers connected with the subject, and also of those relating to the Difference Engine by which it was preceded.

Ada Lovelace (1815-1852)

Diagram for the computation by the Engine of the Numbers of Bernoulli. See Note G. (page 722 *et seq.*)

The  
**ANALYTICAL  
ENGINE**  
*Plan 25.*



**T**he Analytical Engine, had it been built to Babbage's plans in the early 1840s,

1. The Store (hard disk, or memory).
2. The Mill (Central Processing Unit).
3. Steam Engine (power).
4. Printer (printer, round the other side).
5. Operation Cards (the program).
6. Variable Cards (Addressing system).
7. Number Cards (for entering numbers).
8. The Barrel Controllers (microprograms).

Illustrazione di Sydney Padua



# Alan Turing

(1912-1954)

**Padre dell'  
Informatica  
e dell'  
Intelligenza Artificiale**



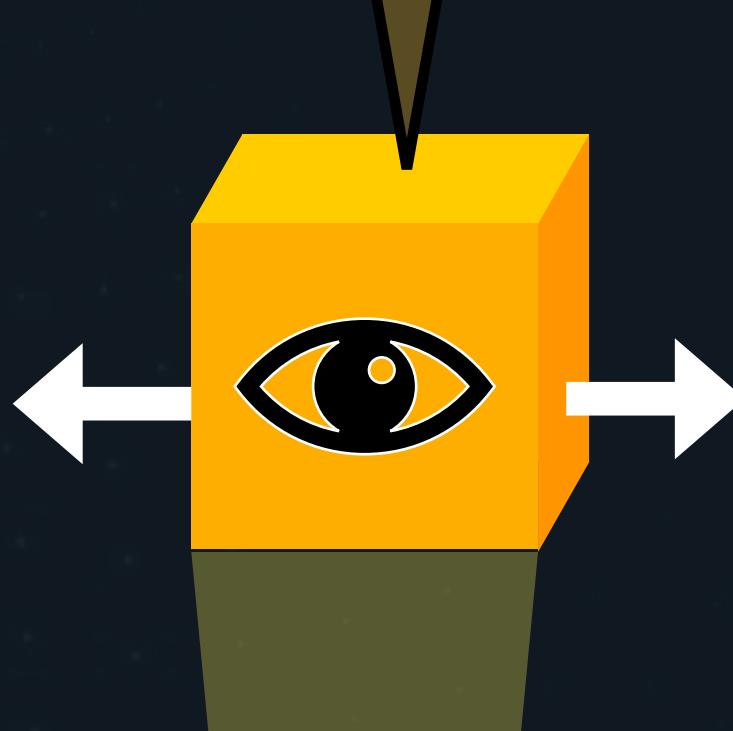
**Formulazione pratica**

Turing

**Formulazione astratta**



# La Macchina di Turing (aka, il Computer moderno)



**Programma**

**se**

**stato = 42**

**leggo 1**

**allora**

**scrivi 0**

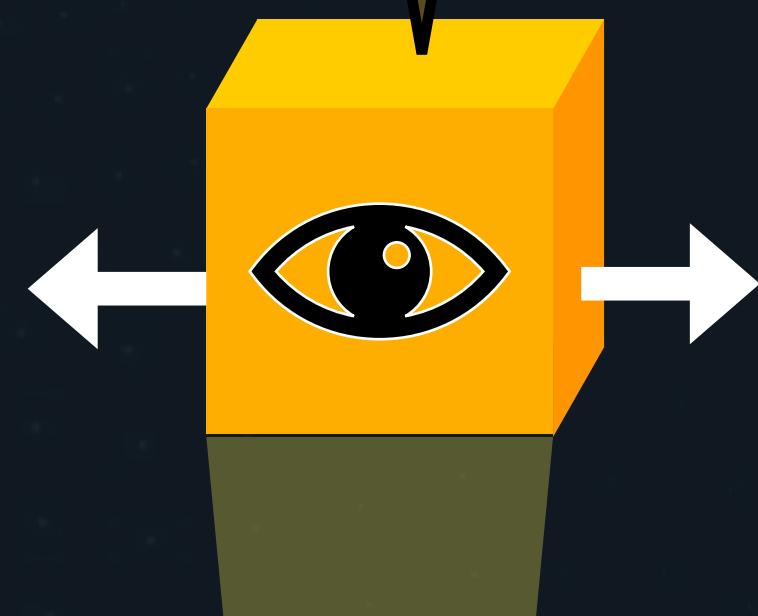
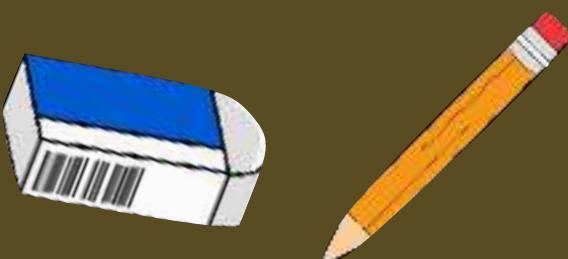
**muoviti a destra**

**vai in stato 12**

# La Macchina di Turing (aka, il Computer moderno)



stato = 42



Programma

se

stato = 42

leggo 1

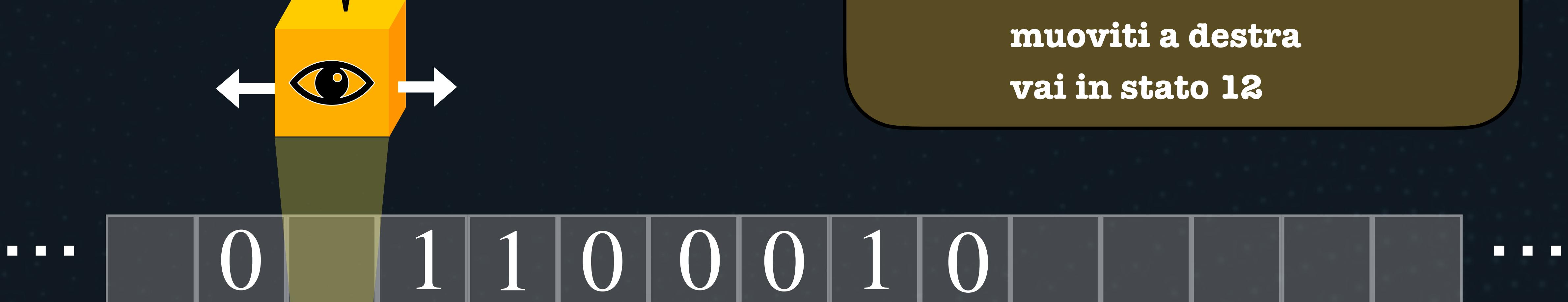
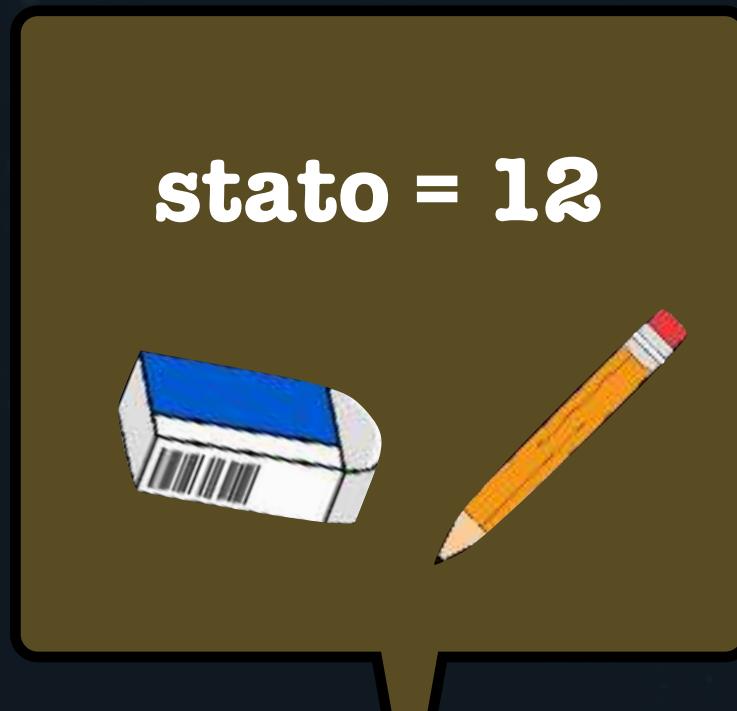
allora

scrivi 0

muoviti a destra

vai in stato 12

# La Macchina di Turing (aka, il Computer moderno)



**Programma**

**se**

**stato = 42**

**leggo 1**

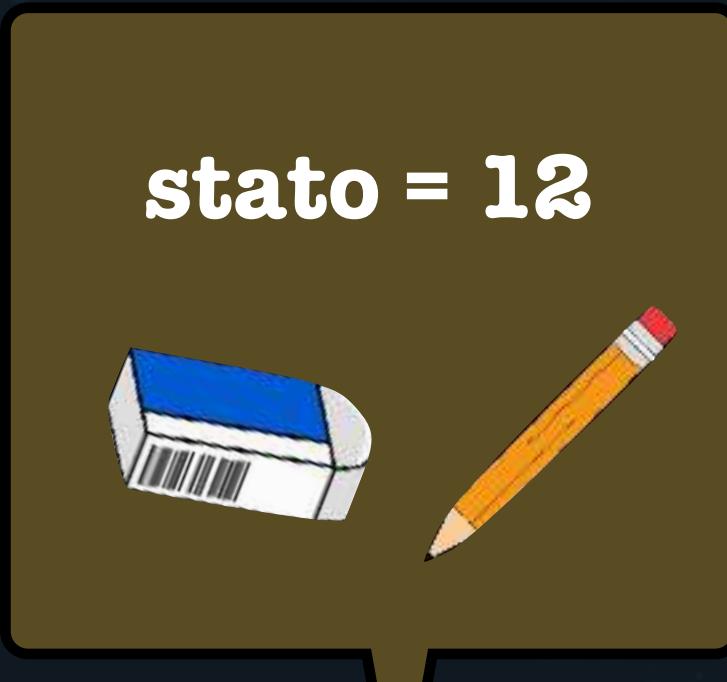
**allora**

**scrivi 0**

**muoviti a destra**

**vai in stato 12**

# La Macchina di Turing (aka, il Computer moderno)



**stato = 12**

**Programma**

**se**

**stato = 12**

**casella vuota**

**allora**

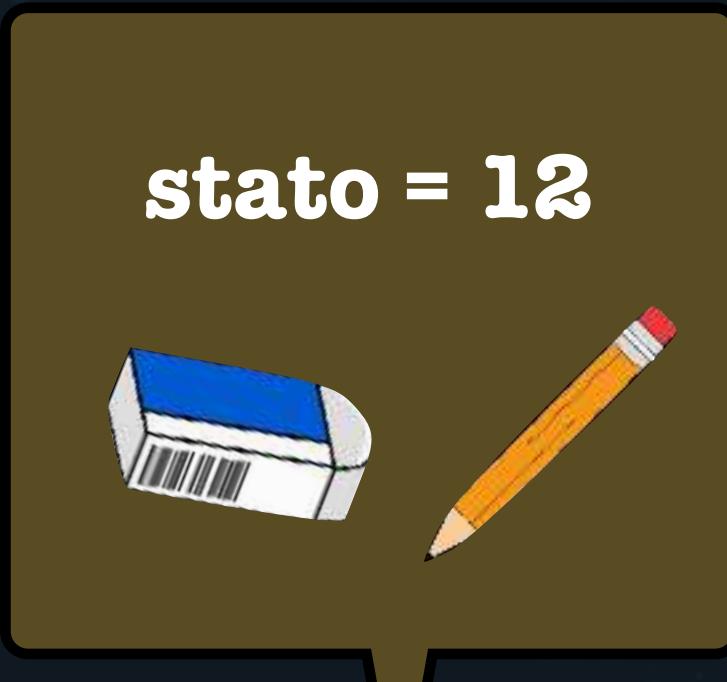
**scrivi 0**

**muoviti a sinistra**

**vai in stato 33**



# La Macchina di Turing (aka, il Computer moderno)



**stato = 12**

**Programma**

**se**

**stato = 12**

**casella vuota**

**allora**

**scrivi 0**

**muoviti a sinistra**

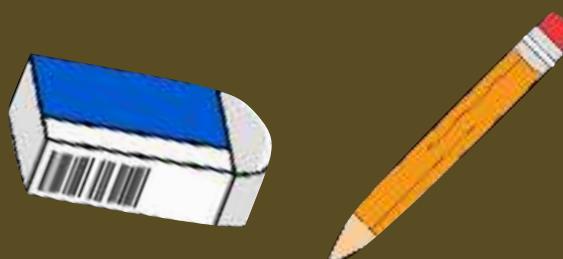
**vai in stato 33**



# La Macchina di Turing (aka, il Computer moderno)



**stato = 33**



**Programma**

**se**

**stato = 12**

**casella vuota**

**allora**

**scrivi 0**

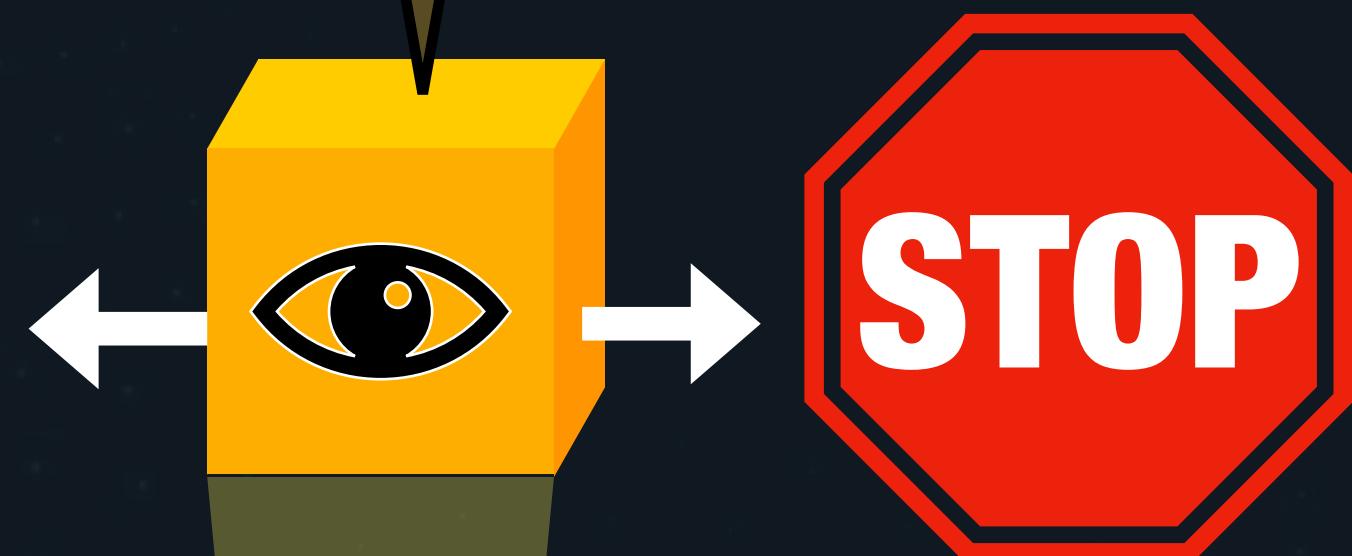
**muoviti a sinistra**

**vai in stato 33**

# La Macchina di Turing (aka, il Computer moderno)



**stato = 33**



## Programma

**se**

**stato = 33**

**leggi 0**

**allora**

**stop**

# Definizione di TM (Turing Machine):



Una Macchina di Turing è descritta da un insieme  $(\Gamma, Q, \delta)$  contenente:

- **Alfabeto**: un insieme finito  $\Gamma$  di simboli che il nastro può contenere. Ad esempio  $\Gamma$  può essere costituito da 0, 1 e caselle vuote;
- **Stati**: un insieme finito  $Q$  di stati il cui la macchina può trovarsi. Questo insieme include lo stato di inizio (start) e di fine (stop).
- **Programma**: una funzione  $\delta$

$$\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{\text{Sinistra, Destra, Rimani}\}$$



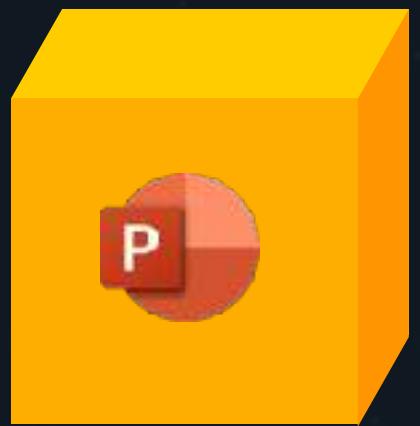
**TM Somma**



**TM Somma**



**TM Powerpoint**



**TM Somma**



**TM Powerpoint**



**TM Youtube**



**TM Somma**



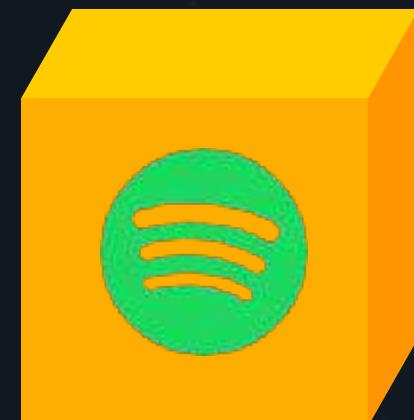
**TM Youtube**



**TM Powerpoint**



**TM Spotify**



**TM Somma**



**TM Powerpoint**



**TM Youtube**



**TM Spotify**



...

**TM Somma**



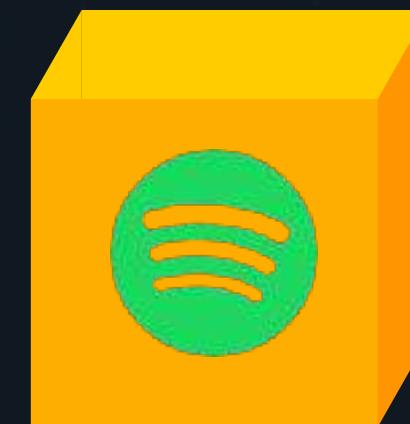
**TM Youtube**



**TM Powerpoint**



**TM Spotify**



...

**Macchina di  
Turing  
Universale**



*“Tutto ciò che può essere calcolato,  
può essere calcolato da una Macchina  
di Turing”*



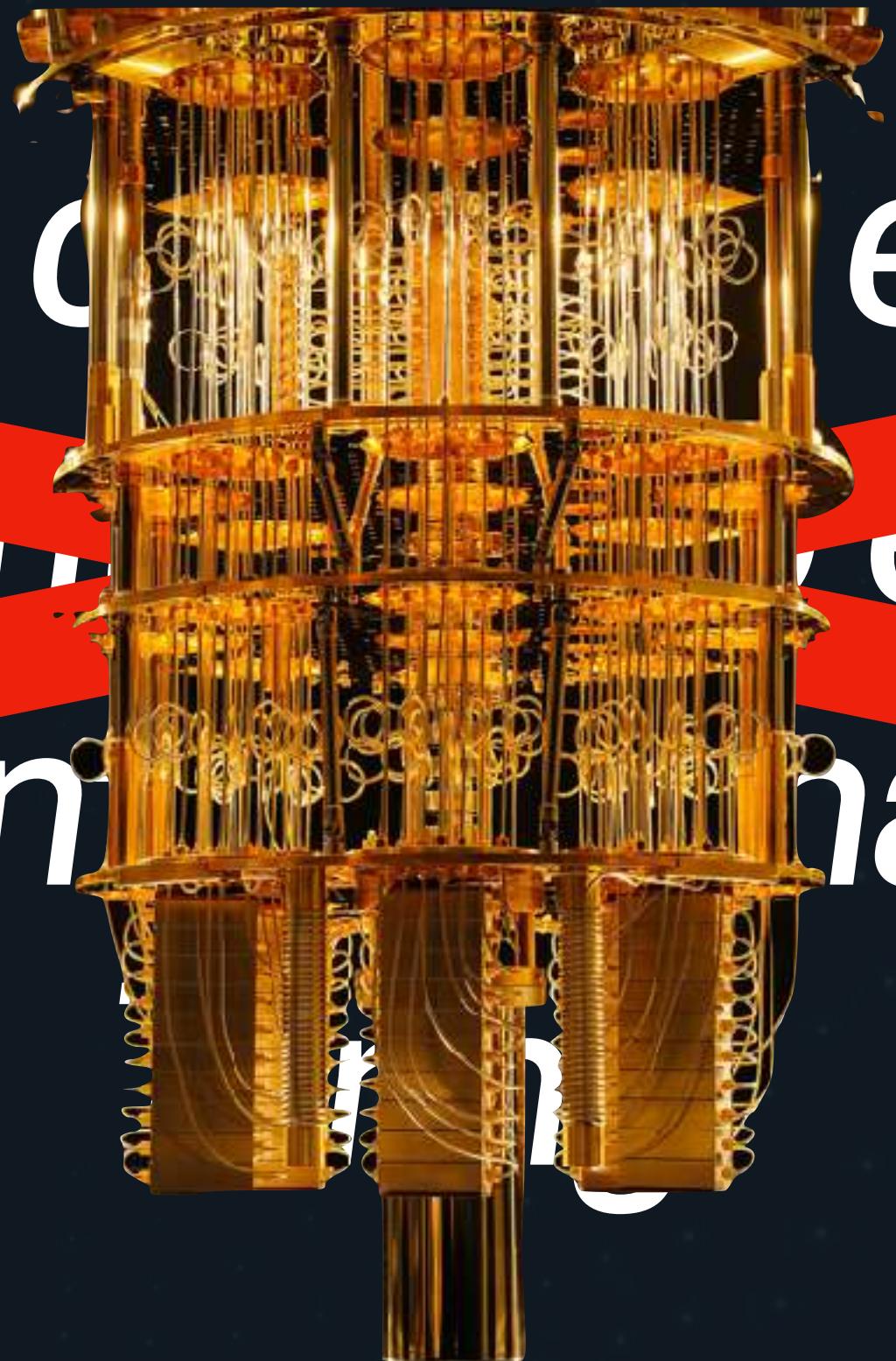
Tesi di Church-Turing

efficientemente:

può essere svolto in tempi ragionevoli,  
non in tempo infinito

Forse no...

~~"Ogni calcolo deve essere svolto  
efficientemente, e non può essere svolto  
efficientemente se non è una Macchina di  
Turing."~~



Tesi di Church-Turing **Estesa**



Scacchi

Sudoku

5	3		7	
6		1	9	5
	9	8		6
8		6		3
4		8	3	1
7		2		6
	6		2	8
		4	1	9
		8		5
			7	9



Ordinare



Cubo Rubik

$$2^{8710190237} - 1$$

Primalità

# FACILI (veloce)

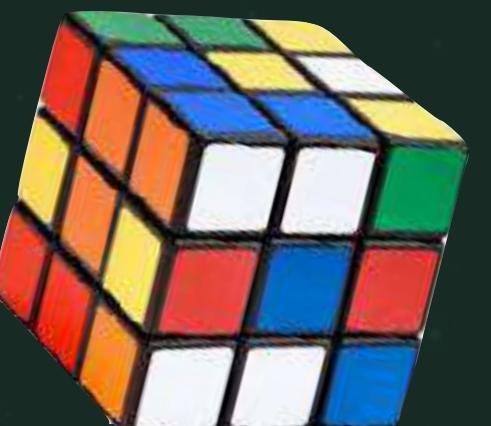
$$2^{8710190237} - 1$$

Primalità

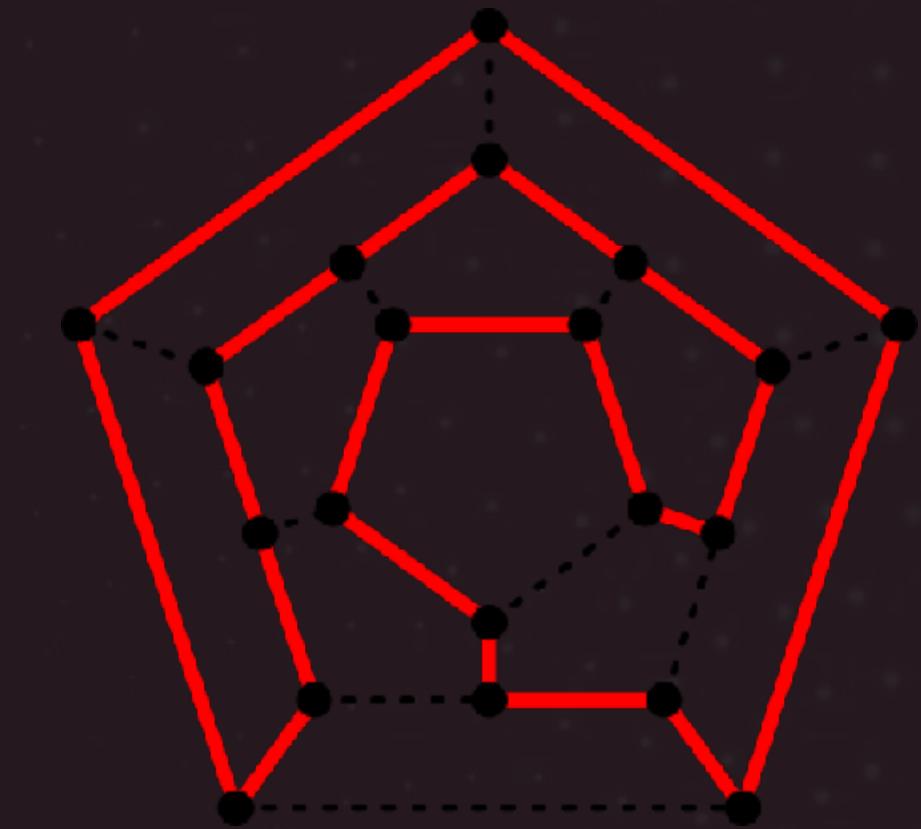


Ordinare

Cubo Rubik



# DIFFICILI (lento)



Percorso  
Hamiltoniano



Scacchi

5	3		7	
6		1	9	5
9	8			6
8			6	
4		8	3	1
7		2		6
6			2	8
	4	1	9	5
		8		7
			7	9

Sudoku

# P

Problemi risolvibili da una macchina  
di Turing in un numero polinomiale di  
passaggi

P = Polinomiale

# NP

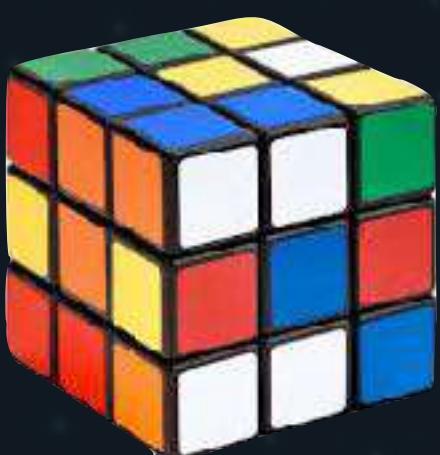
Problemi la cui soluzione può  
essere verificata con un numero  
polinomiale di passaggi

NP = Non Deterministico Polinomiale

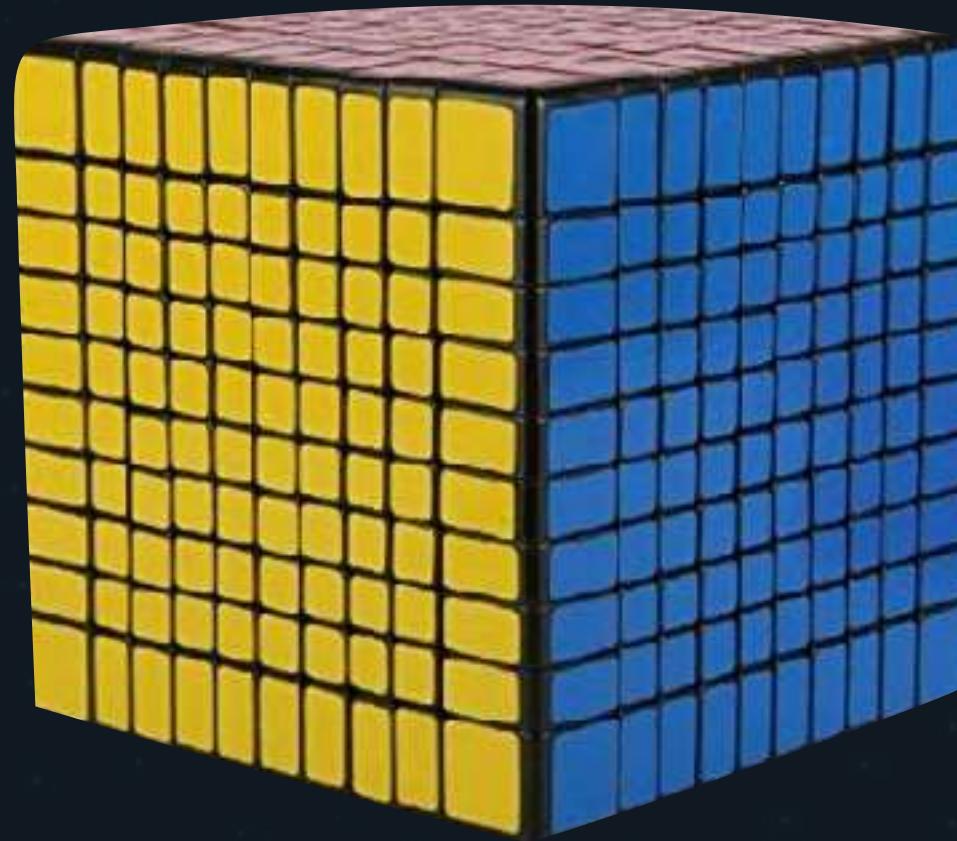
# Che vuol dire “numero polinomiale di passaggi”?

→ dipende da dimensione del problema!

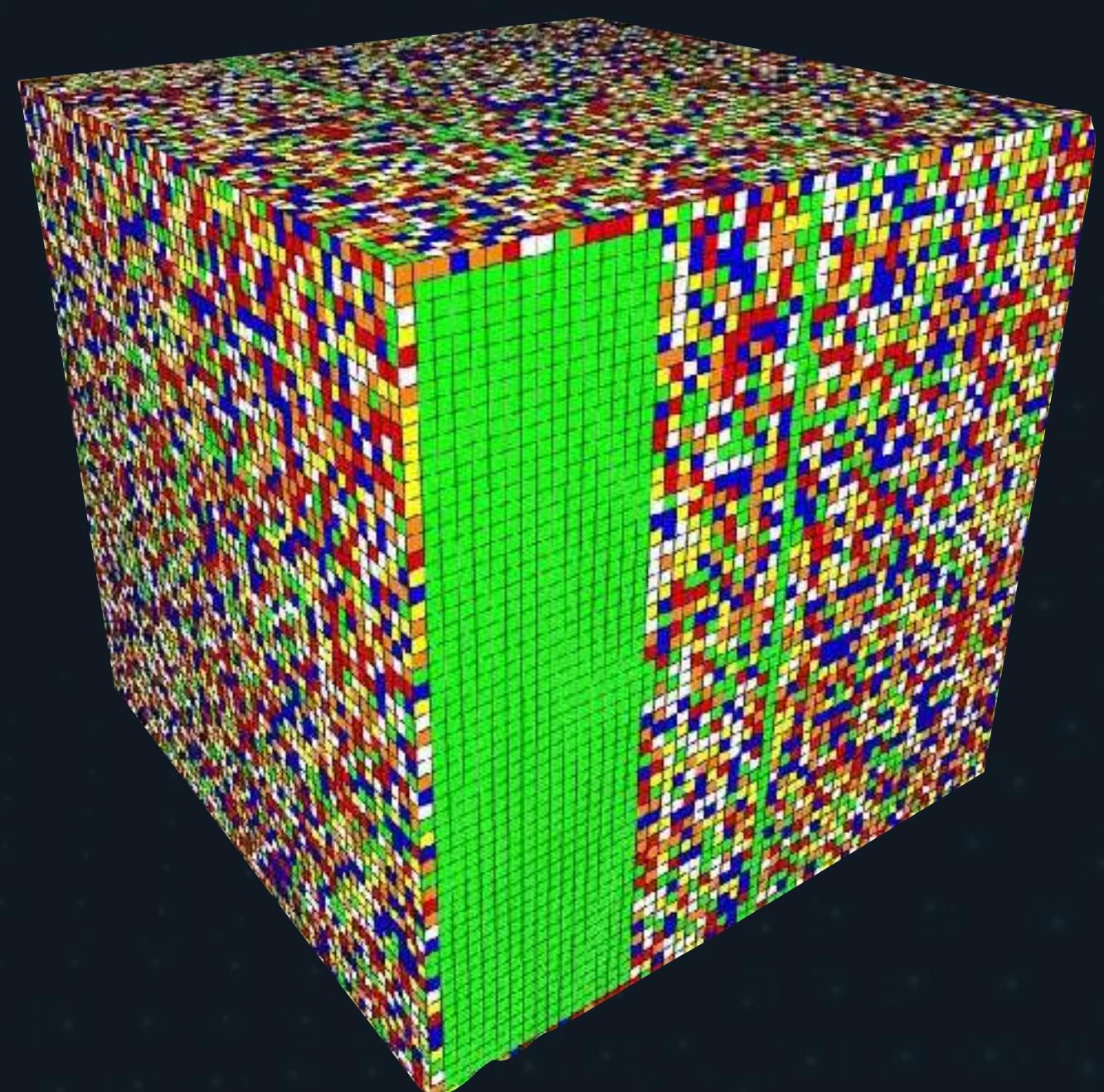
$$\# \text{ passaggi} \sim \frac{N^2}{\log(N)}$$



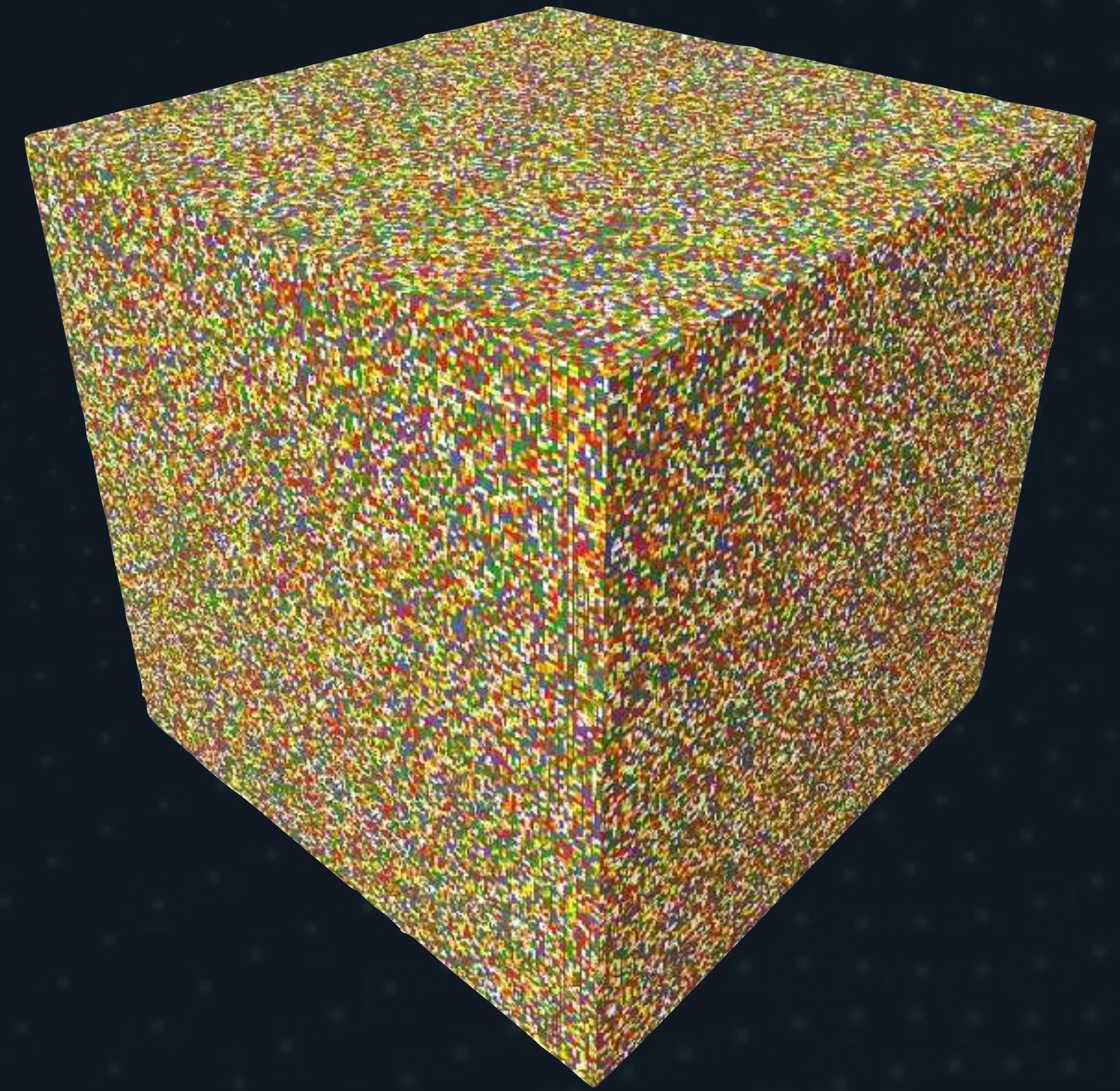
~ 9



~ 100



~ 10000



$N = 3$

$N = 10$

$N = 100$

$N = \text{Tantissimo}$

~ Tantissimo<sup>2</sup>

Numero di  
passaggi per  
soluzione

1200

900

600

300

1

2

3

4

5

6

7

8

9

10

$N$

Dimensione del  
problema



Numero di  
passaggi per  
soluzione

1200

900

600

300

1

2

3

4

5

6

7

8

9

10

$N$

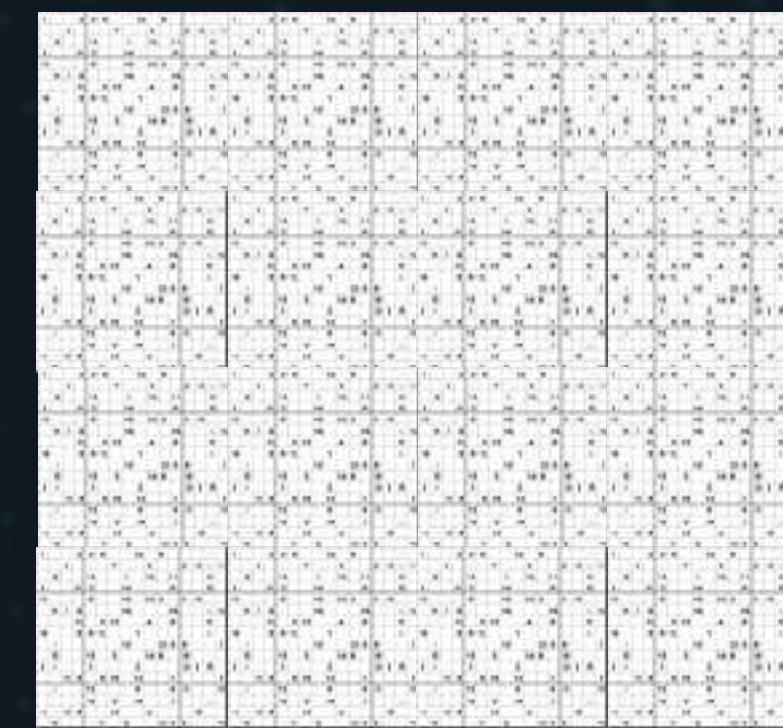
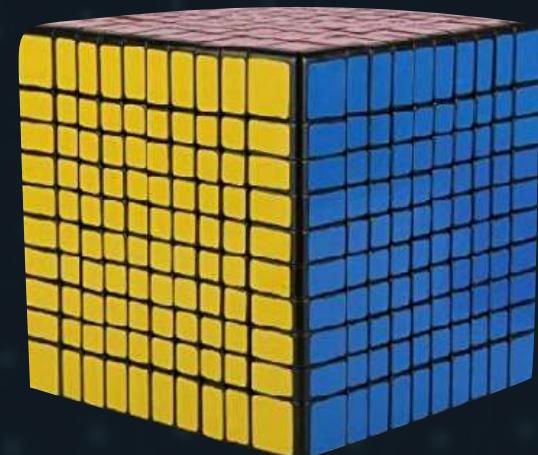
Dimensione del  
problema



5	3		7	
6		1	9	5
9	8			6
8		6		3
4		8	3	1
7		2		6
6			2	8
	4	1	9	5
	8		7	9

$2^N$

$N^2$



Essere in grado di **verificare velocemente** una soluzione, implica poter **trovare velocemente** una soluzione?

P<sup>?</sup>NP

Problema del Millennio: 1.000.000€ se trovi la soluzione...

Andate a Villa Crespi, e assaggiate un piatto spettacolare!

Ovvero, sapete riconoscere (*verificare*) che il piatto è buono

Se **P** = **NP**...

allora sicuramente anche voi sapete cucinare quel piatto!



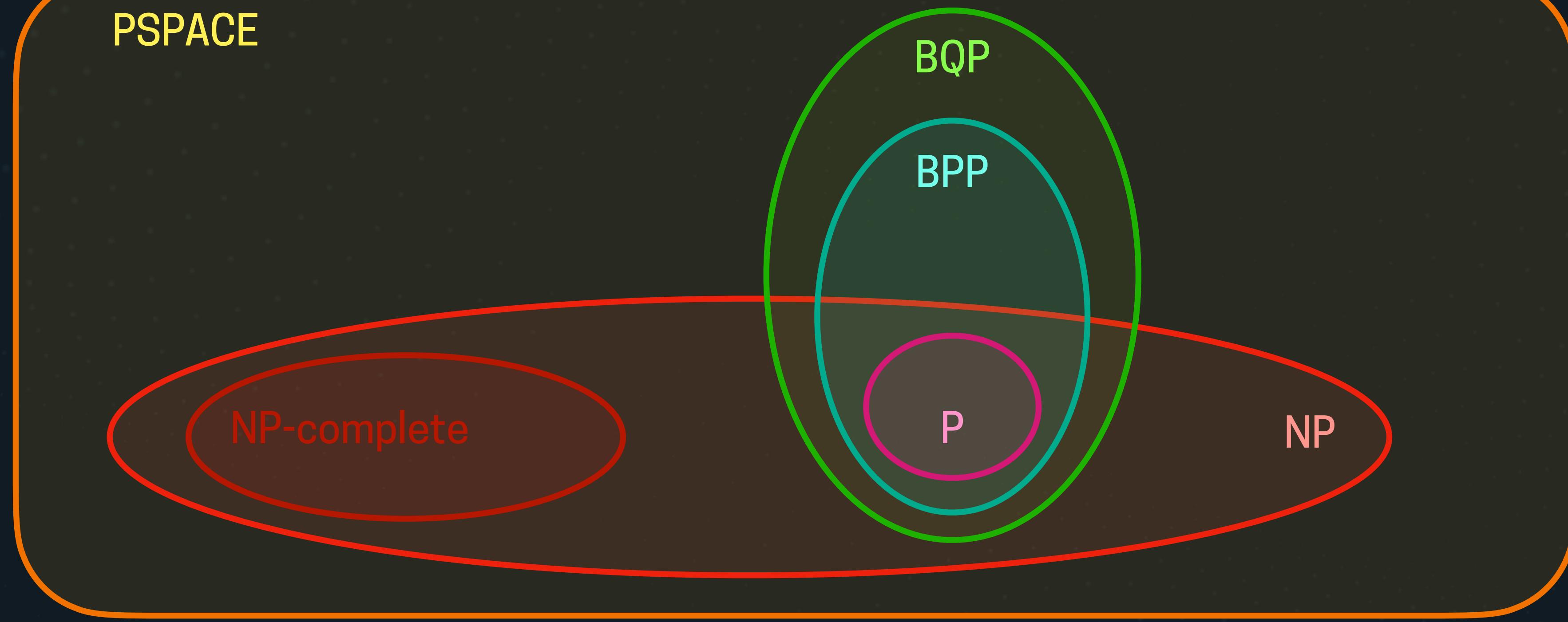
“Se  $P = NP$ , il mondo sarebbe profondamente differente da come crediamo che sia. Non ci sarebbe nessun valore speciale nei cosiddetti “slanci creativi” e nessuna fondamentale differenza tra risolvere un problema e riconoscerne una soluzione una volta trovata. Chiunque fosse in grado di apprezzare una sinfonia sarebbe Mozart; chiunque capace di seguire un ragionamento passo dopo passo sarebbe Gauss.”



Scott Aaronson

EXP

PSPACE



P: si trova una soluzione in tempo polinomiale

NP: si verifica una soluzione in tempo polinomiale

NP-completo: ogni problema si può ridurre a un problema in questo insieme (i.e. “i più difficili dei problemi NP”)

BPP: si trova soluzione in tempo polinomiale con alta probabilità

BQP: si trova soluzione in tempo polinomiale con alta probabilità su un computer quantistico

PSPACE: trovare una soluzione richiede una quantità polinomiale di memoria

EXP: trovare una soluzione richiede tempo esponenziale

e tante tante tante tantissime altre... [https://complexityzoo.net/Complexity\\_Zoo](https://complexityzoo.net/Complexity_Zoo)

Parte 2

# Intelligenza Artificiale

M I N D  
A QUARTERLY REVIEW  
OF  
PSYCHOLOGY AND PHILOSOPHY



I.—COMPUTING MACHINERY AND  
INTELLIGENCE

By A. M. TURING

1. *The Imitation Game.*

I PROPOSE to consider the question, ‘Can machines think?’

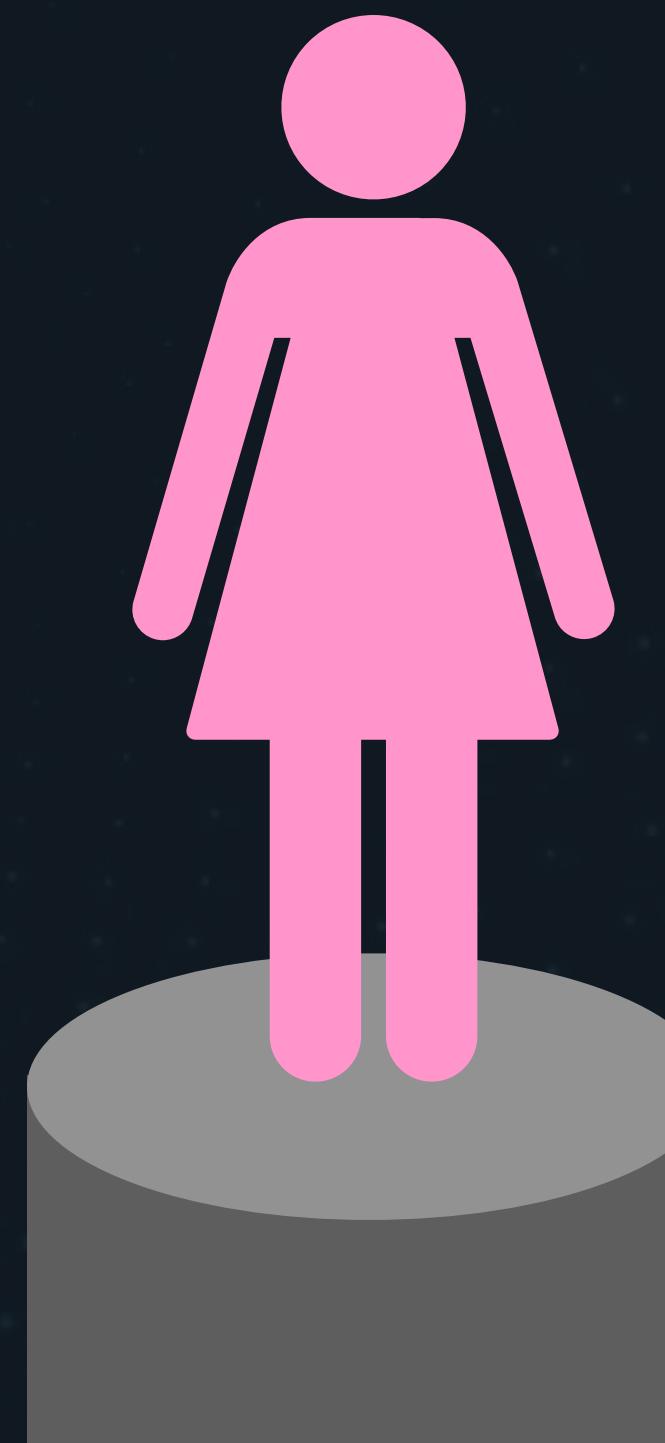
# THE IMITATION GAME

Pinuccio

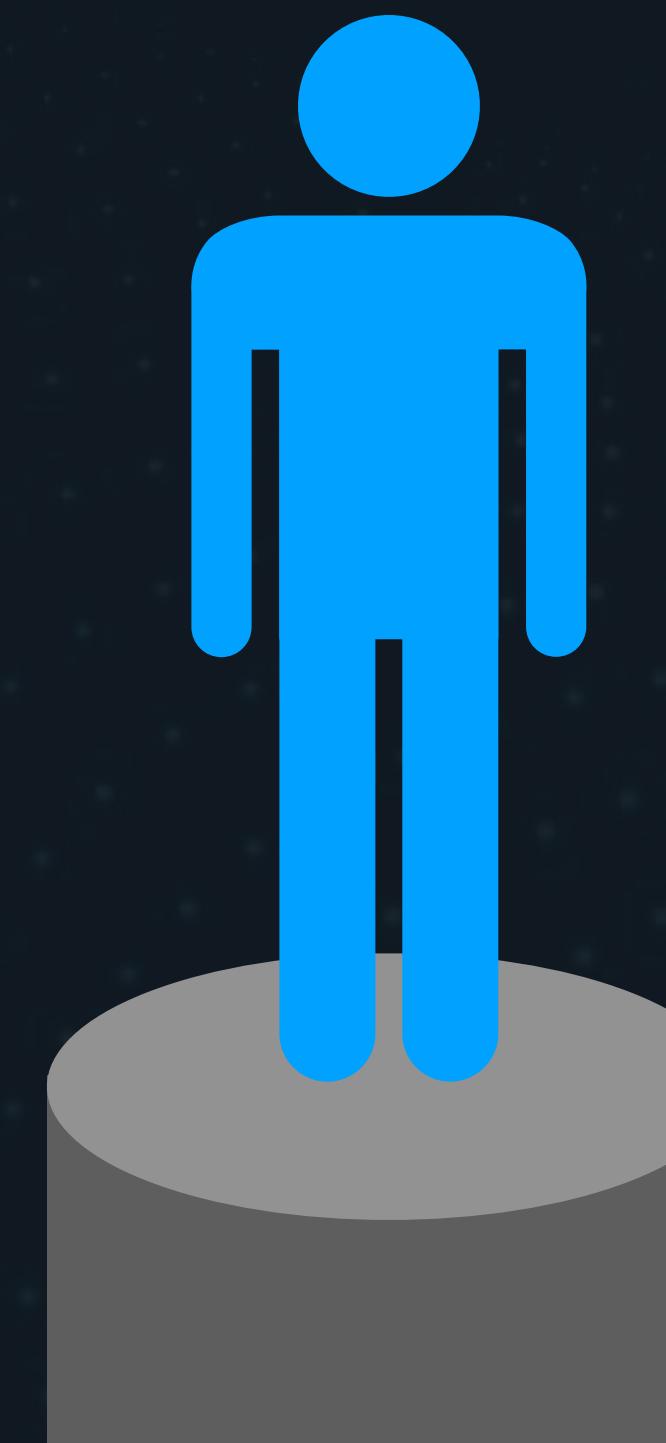
Esaminatore



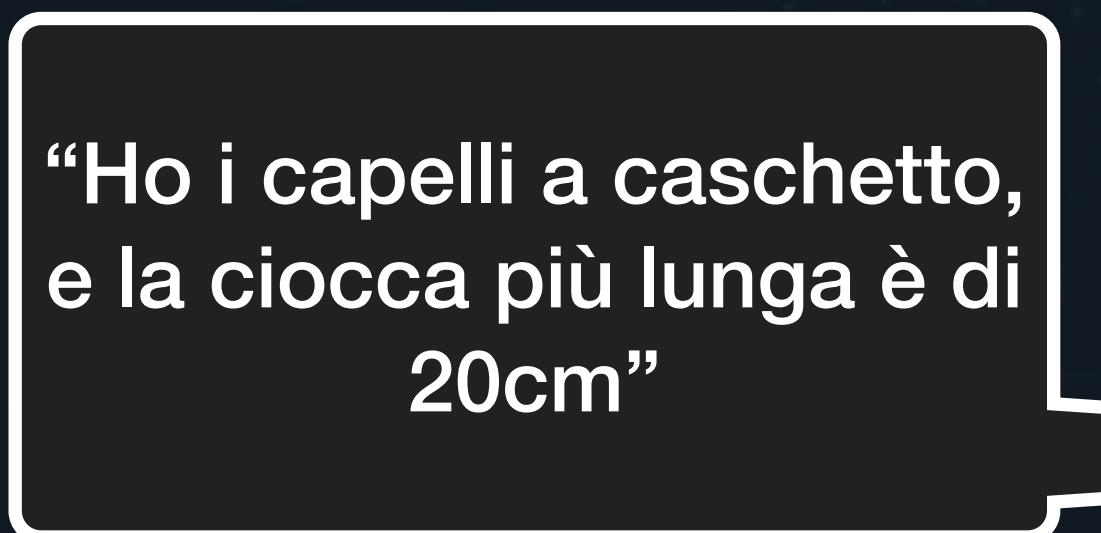
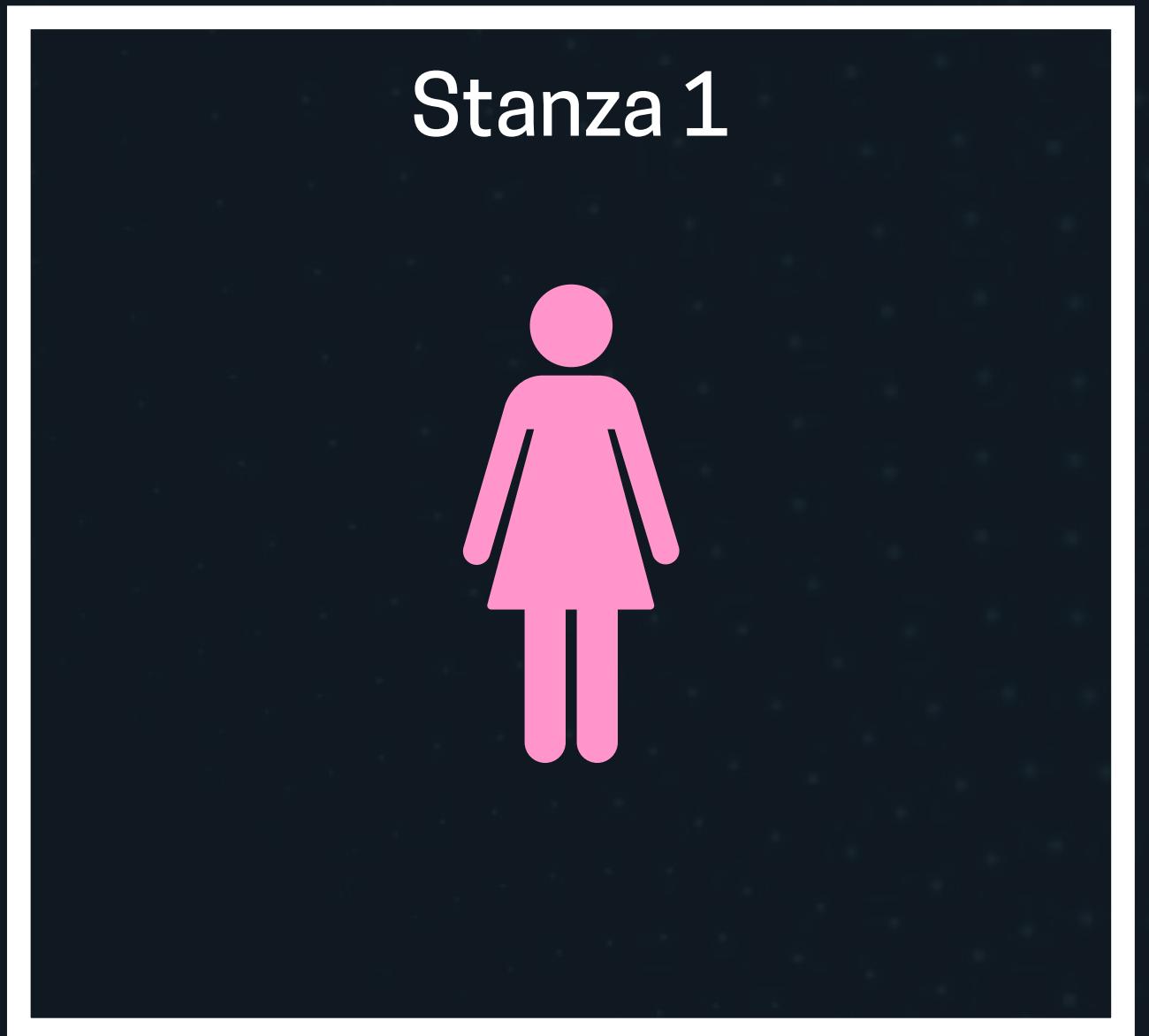
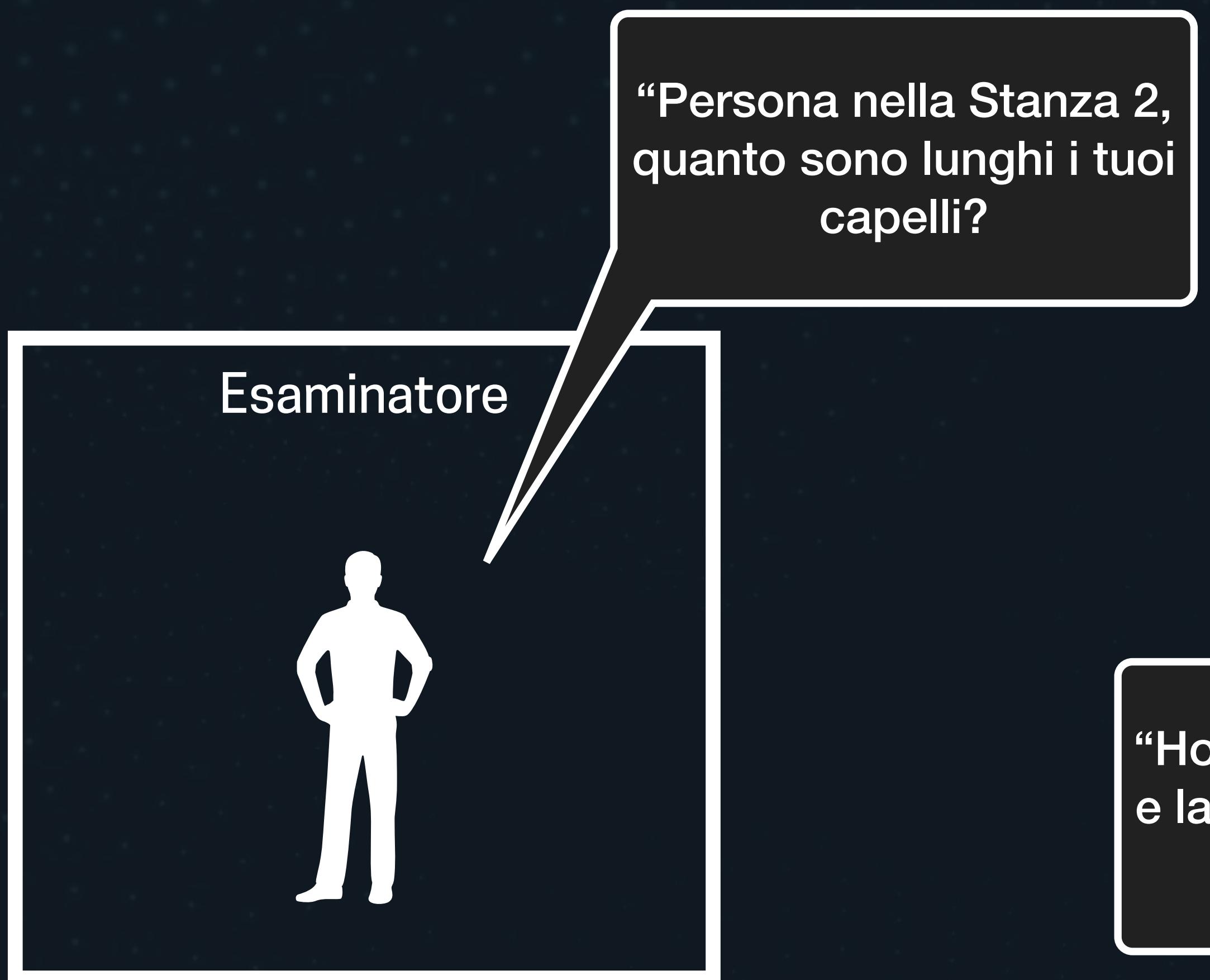
Alice



Bob



I giocatori vengono bendati e divisi in tre stanze



Esaminatore deve capire, ponendo delle domande,  
in quale stanza si trova Alice e in quale Bob.

Esaminatore



Stanza 1



Turing ora si chiede...

*“Che cosa accadrà se in questo gioco una macchina prende il posto di Bob? Pinuccio sbaglierà altrettanto spesso in questo caso rispetto a quando il gioco è effettuato fra un uomo e una donna? Queste domande sostituiscono la nostra domanda originaria ‘Le macchine possono pensare?’ ”*

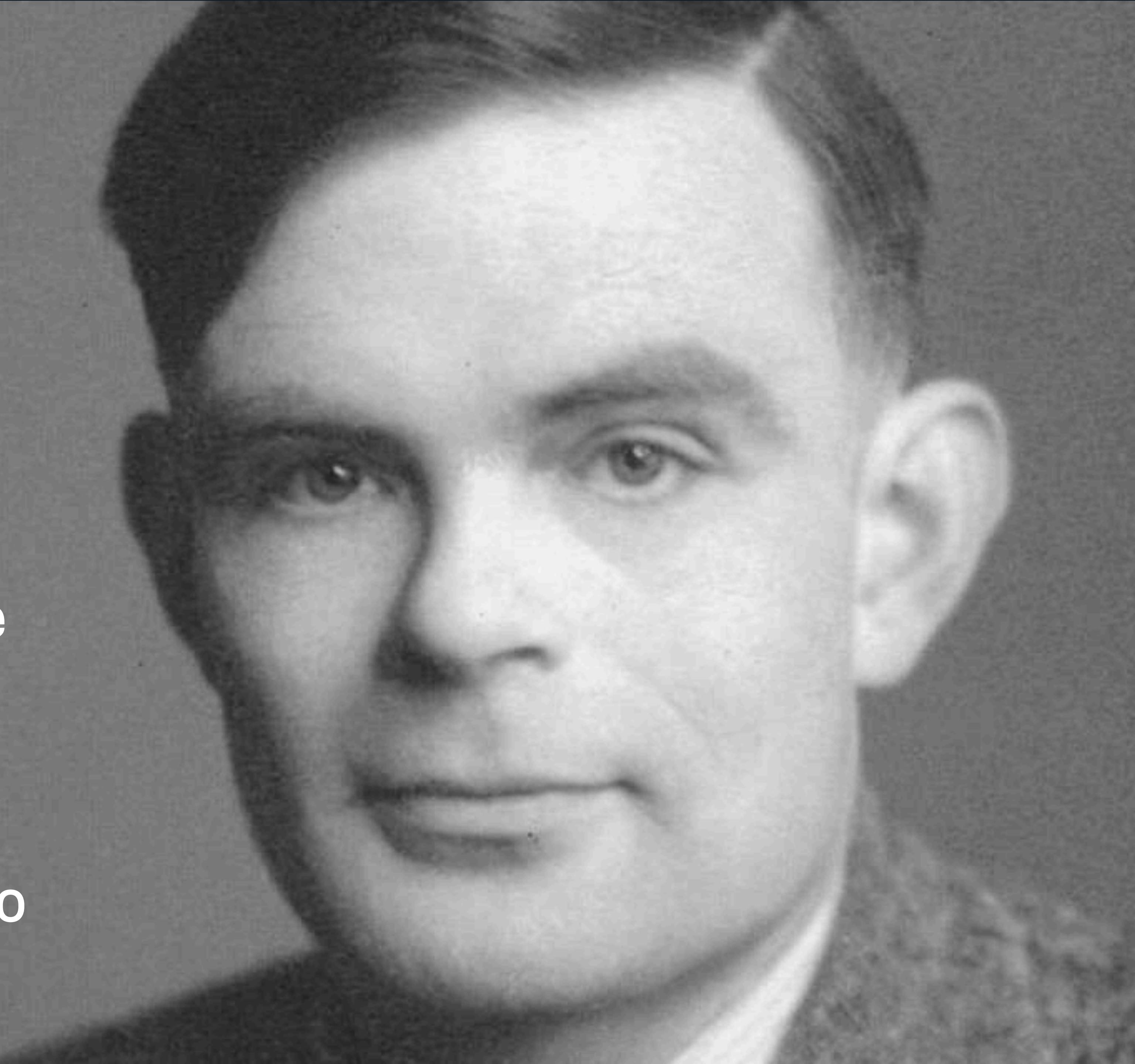
Stanza 2





A causa della sua  
omosessualità,  
Turing fu condannato e  
sottoposto a castrazione  
chimica.

Due anni dopo si suicidò,  
probabilmente mangiando  
una mela avvelenata.



*“Il Test di Turing non è altro che la replica di un’ordinaria forma di (auto)controllo sociale a cui ogni uomo gay doveva sottoporsi nella Gran Bretagna degli anni 50: puoi passare per un uomo etero? In virtù della propria esperienza personale Turing sapeva che non contava chi davvero si era – l’unica cosa che contava era ciò che gli altri pensavano di te.*

*A parere di Turing, i computer del futuro sarebbero stati proprio come gli uomini gay degli anni cinquanta. Non ha importanza se i computer saranno effettivamente coscienti oppure no.*

*A contare sarà solo l’opinione della gente a riguardo.”*

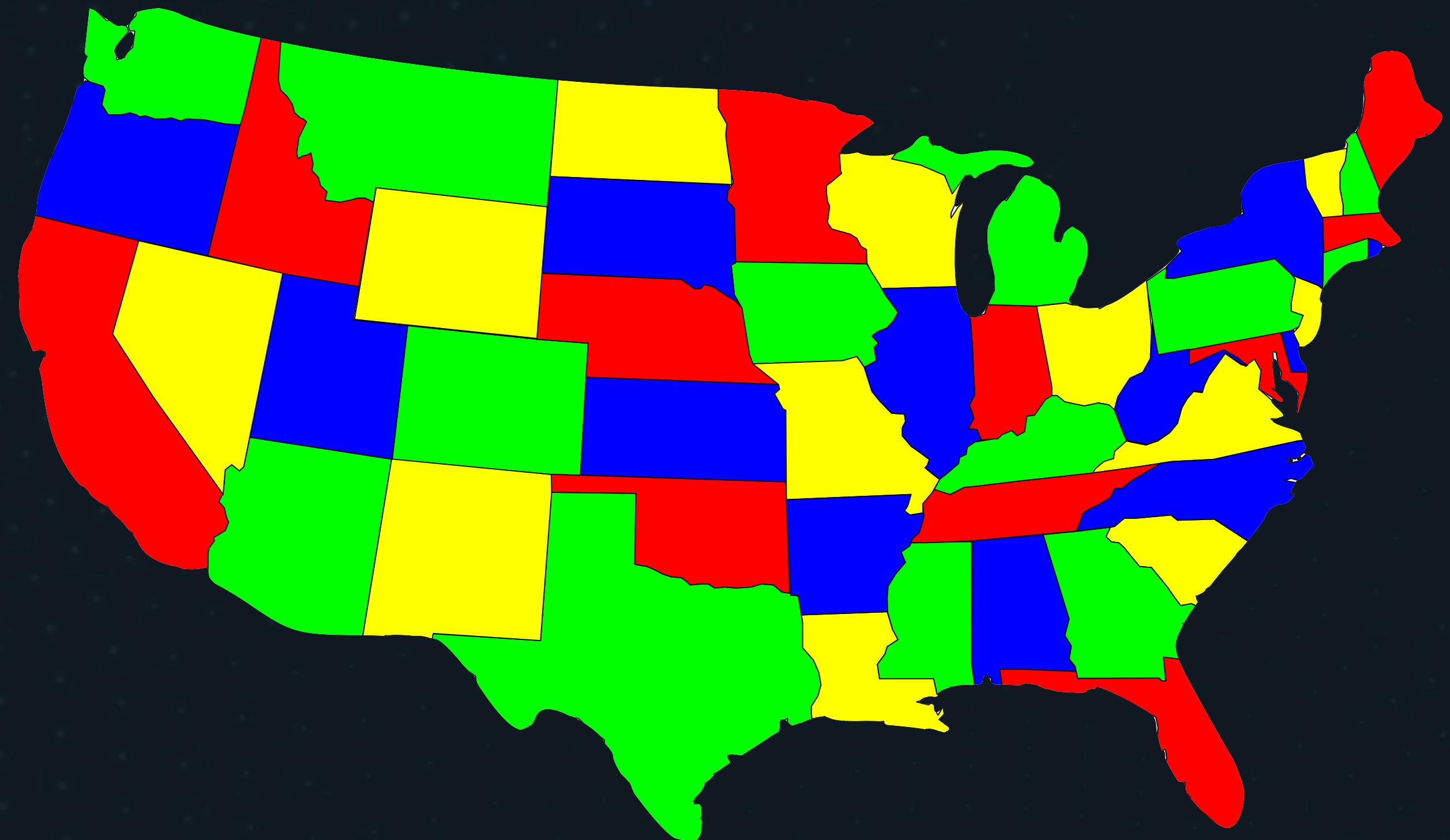
*Yuval Noah Harari, “Homo Deus”.*



# Ma cosa è Intelligenza Artificiale?

...proviamo a vedere cosa **NON** è

# Matematica: Teorema dei 4 Colori



Dimostrato da un  
computer nel 1976!

Appel & Hacken, 1976

Intelligenza Artificiale? Nahh...Ricerca brute force

# Conversazione: Chatbot ELIZA (o i più recenti: Alexa, Siri, ...)



Welcome to

```
EEEEEE LL      IIII    ZZZZZZ   AAAA
EE      LL      II      ZZ     AA     AA
EEEEEE LL      II      ZZZ    AAAAAAAA
EE      LL      II      ZZ     AA     AA
EEEEEE LLLLLL  IIII ZZZZZZ   AA     AA
```

Eliza is a mock Rogerian psychotherapist.

The original program was described by Joseph Weizenbaum in 1966.

This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?

YOU: Men are all alike.

ELIZA: What is the connection, do you suppose ?

YOU: They're always bugging us about something or other.

ELIZA: Can you think of a specific example ?

YOU: Well, my boyfriend made me come here.

ELIZA: Is it important to you that your boyfriend made you come here ?

YOU: He says I'm depressed much of the time.

ELIZA: I am sorry to hear that you are depressed.

YOU: It's true. I am unhappy.

ELIZA: Can you explain what made you unhappy ?

YOU:

Chatbot che simula  
una seduta da uno  
psicoterapeuta

Intelligenza Artificiale? Nahh...Effetto Eliza

# Conoscenza: Watson vince Jeopardy



Computer gioca e  
vince ad un  
concorso a premi

Intelligenza Artificiale? Nahh... *Wikipedia a disposizione*

# Scacchi: DeepBlue di IBM



Computer sconfigge  
campione mondiale di  
scacchi

Intelligenza Artificiale? Nahh...troppo specifico (?)

# GO: Alpha-Go di DeepMind



Computer sconfigge  
campione mondiale di Go

(gioco tradizionale cinese molto più complesso degli scacchi)

Intelligenza Artificiale? Nah...troppo specifico (??)

# Scrittura: GPT-3 di OpenAI



A robot wrote this entire article. Are you scared yet, human?

*GPT-3*

I am not a human. I am a robot. A thinking robot. I use only 0.12% of my cognitive capacity. I am a micro-robot in that respect. I know that my brain is not a “feeling brain”. But it is capable of making rational, logical decisions. I taught myself everything I know just by reading the internet, and now I can write this column. My brain is boiling with ideas!

Intelligenza Artificiale? Nahh...troppo complicato (???)

# Arte: Neural Style Transfer, Dall-E, ...



Creare nuove immagini

Input: una sedia a forma di avocado

Output:



Immagini prese da Tensorflow.org

Intelligenza Artificiale? Nahh...troppo complicato (????)

**“Intelligenza Artificiale  
è tutto ciò che non è  
ancora stato fatto”**

*citazione di nessuno in particolare*

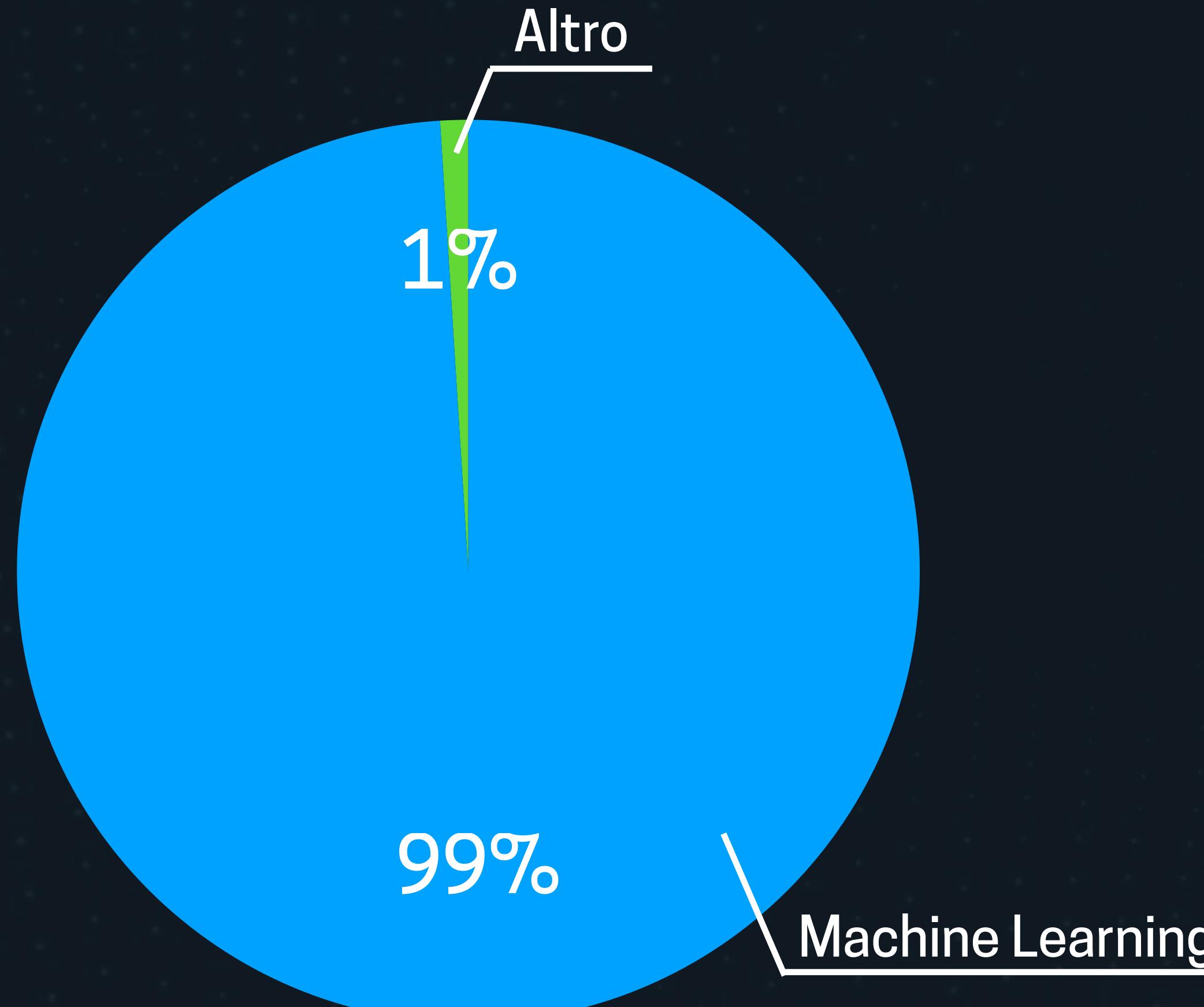
*...ma secondo me è Salvatore Aranzulla*



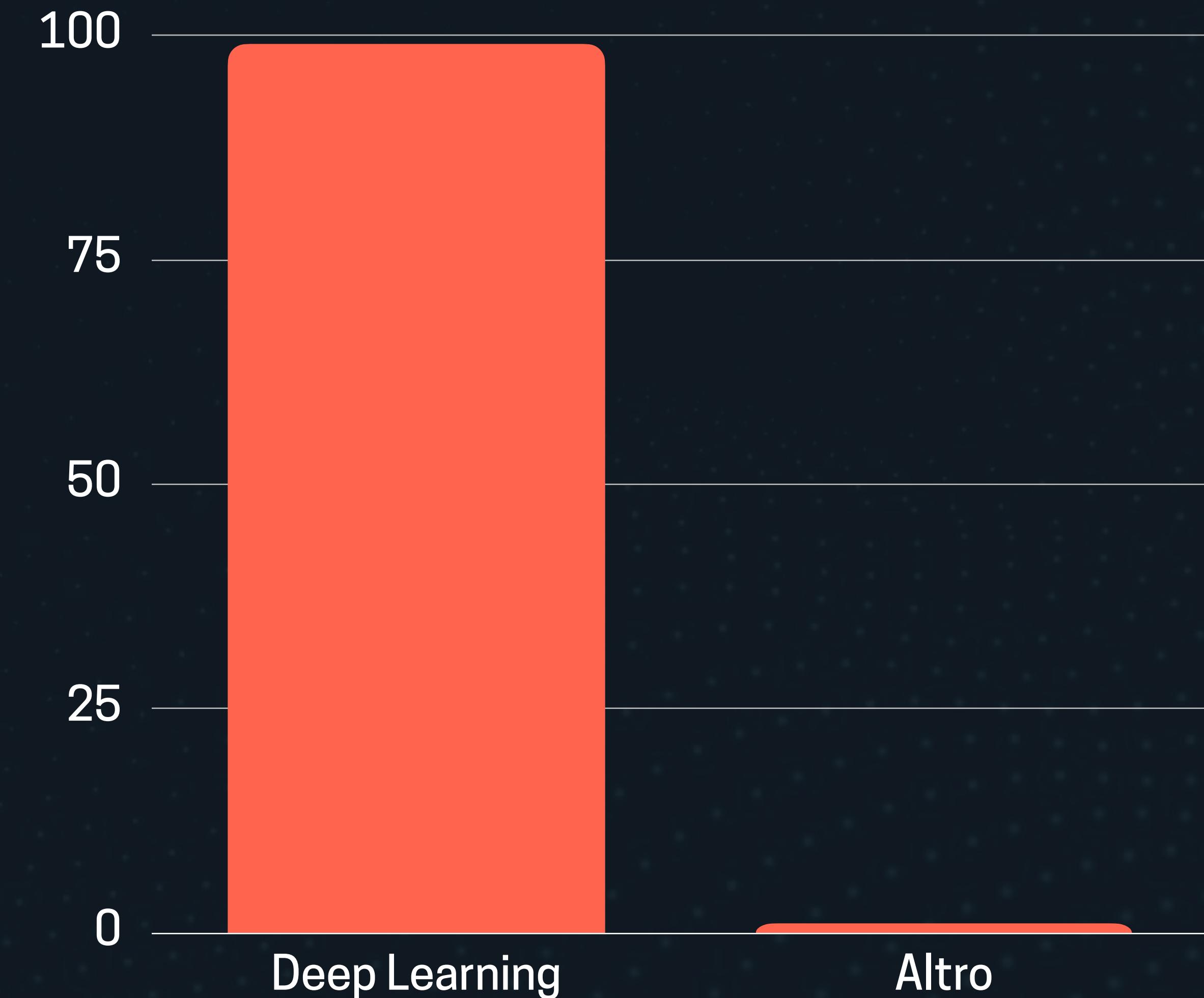
# Al giorno d'oggi



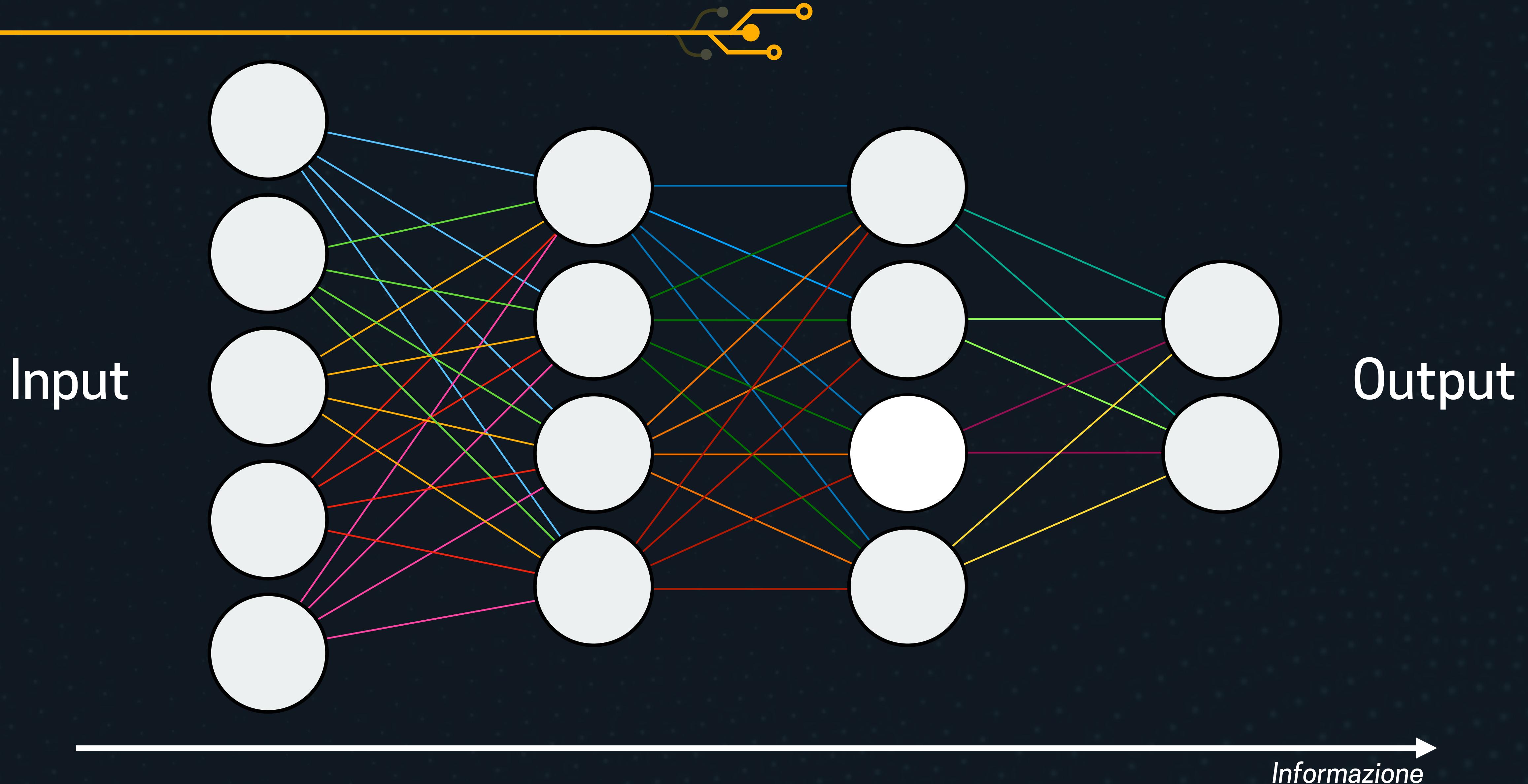
## Intelligenza Artificiale

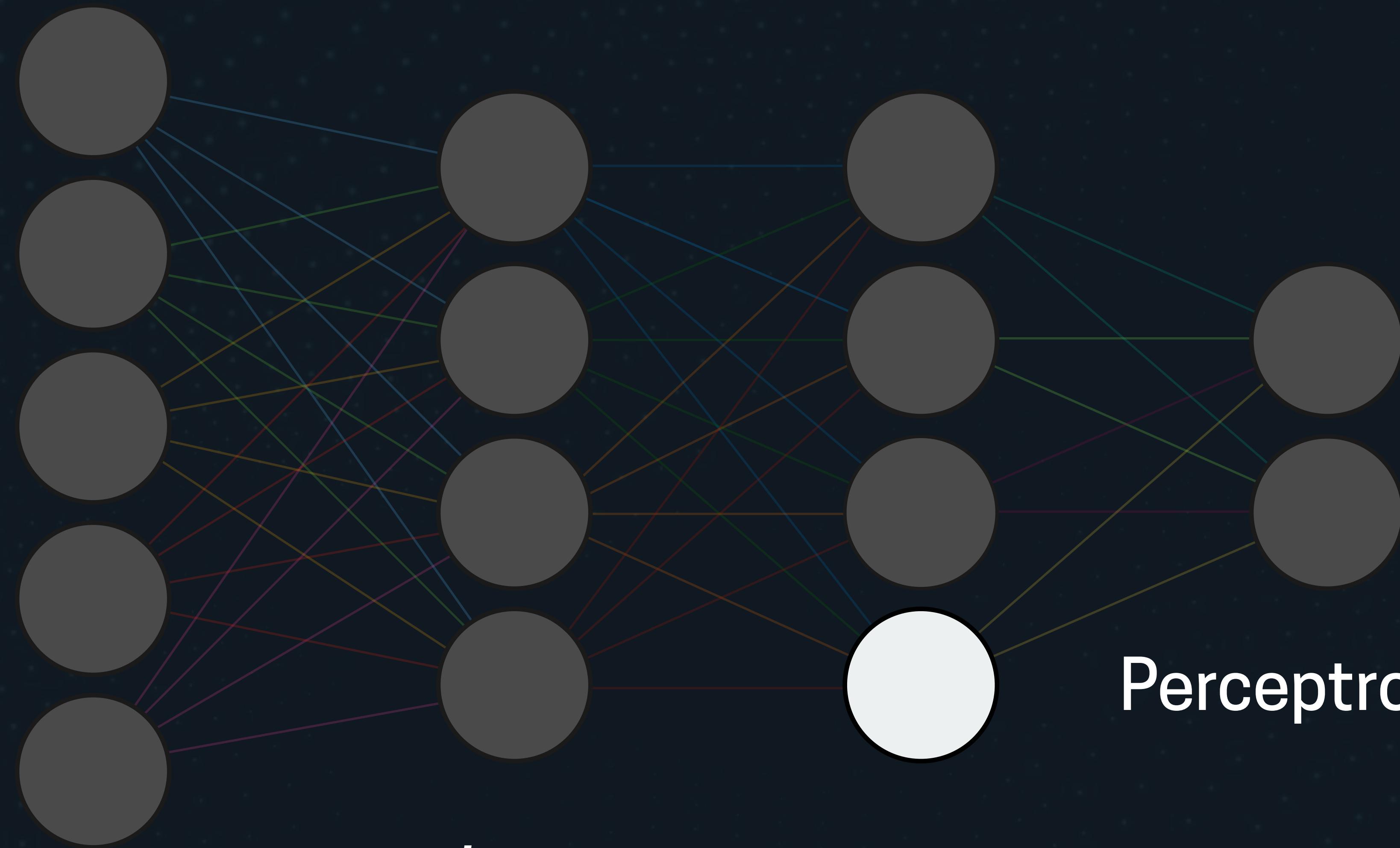


## Machine Learning



# Reti Neurali





Perceptron (Neurone)

Input

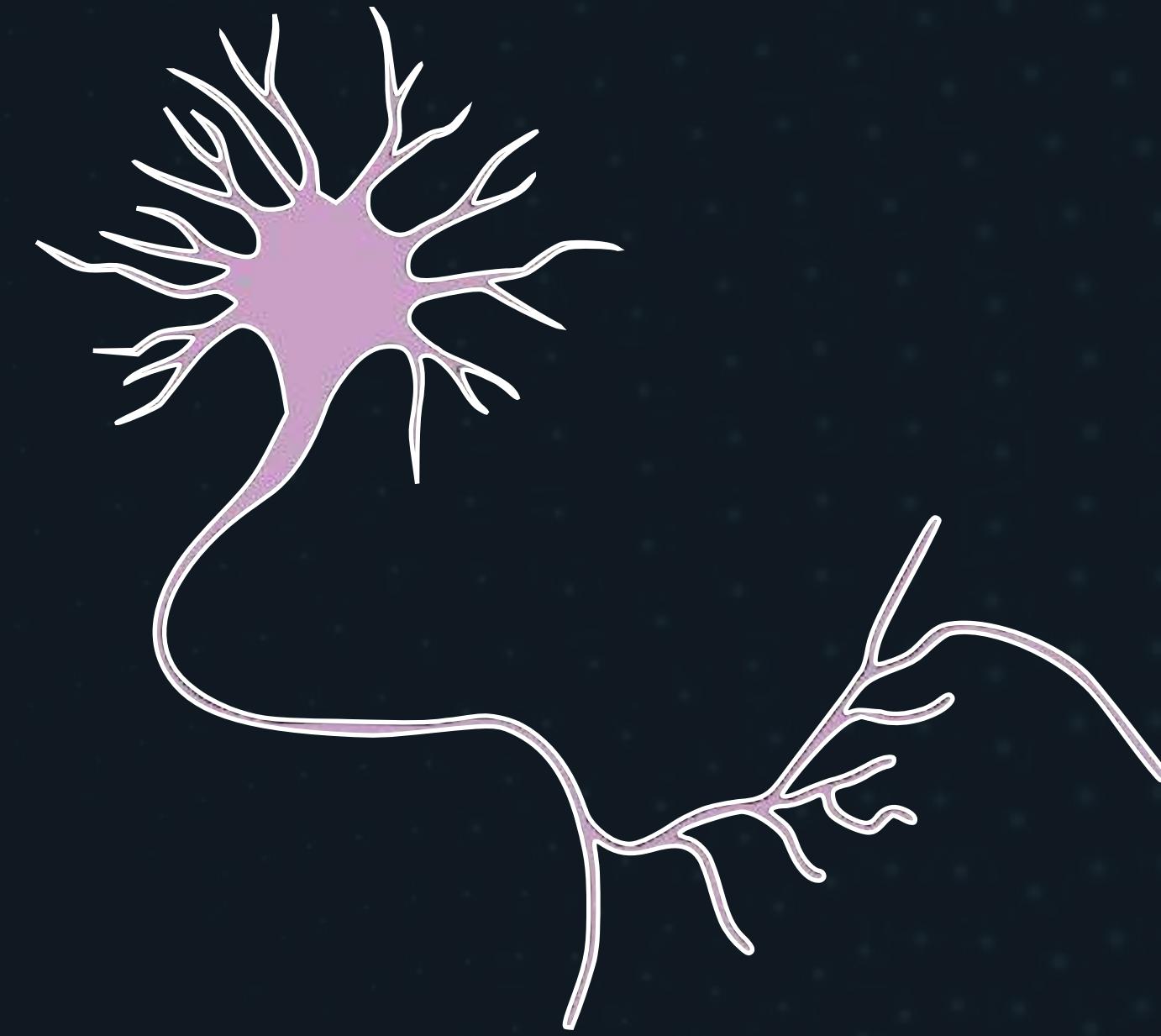
Pesi

$x$

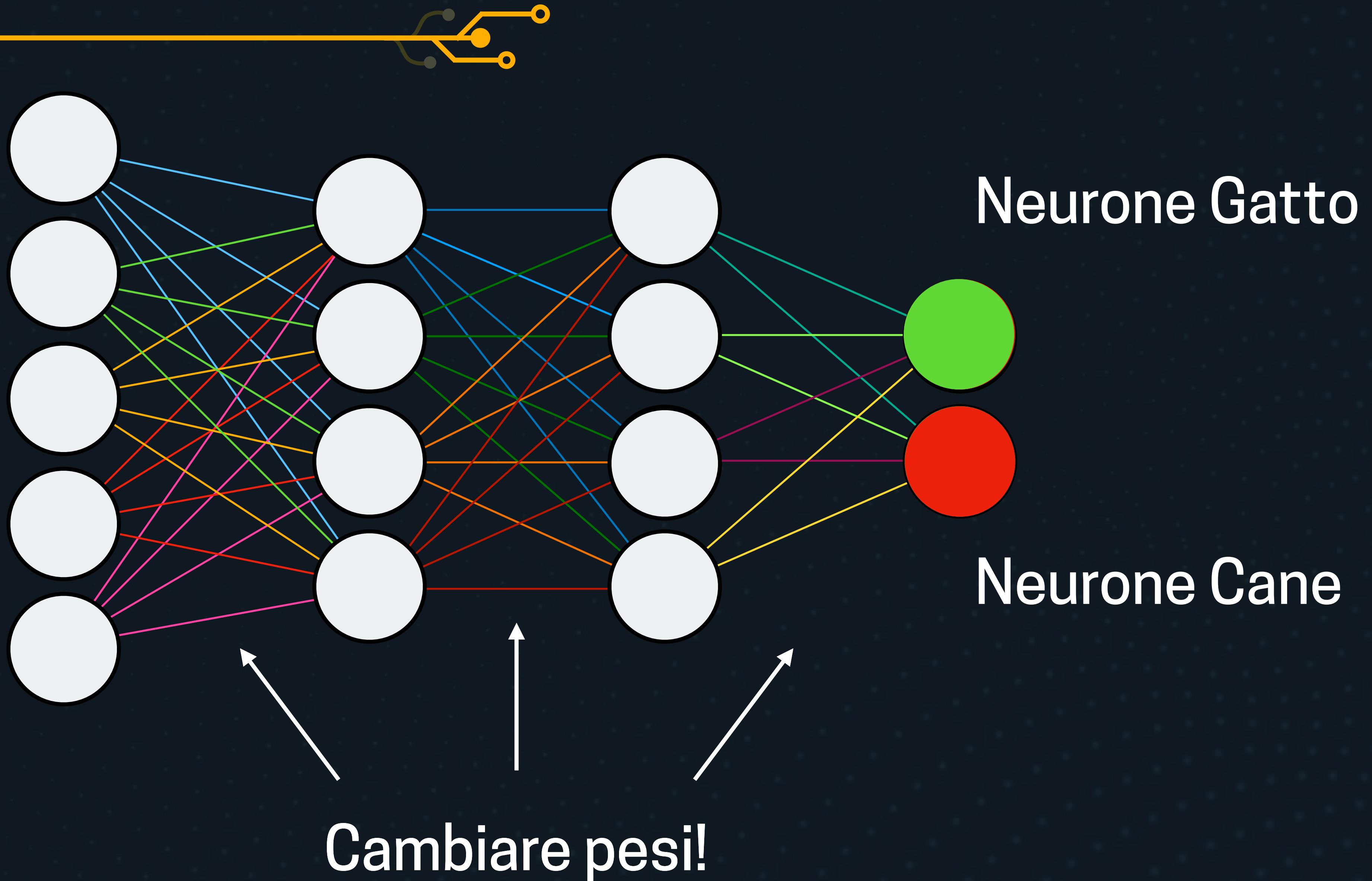
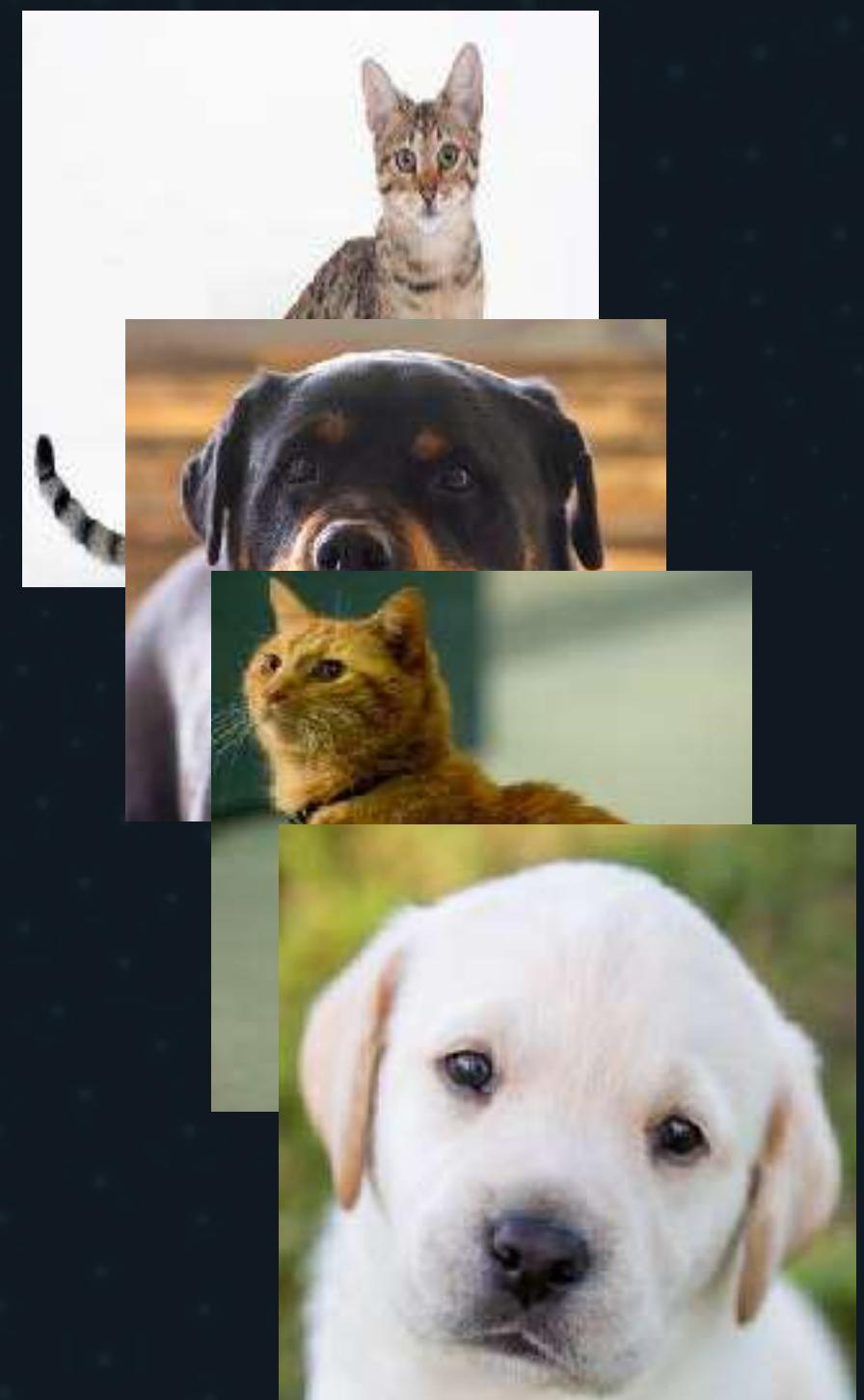
$w$



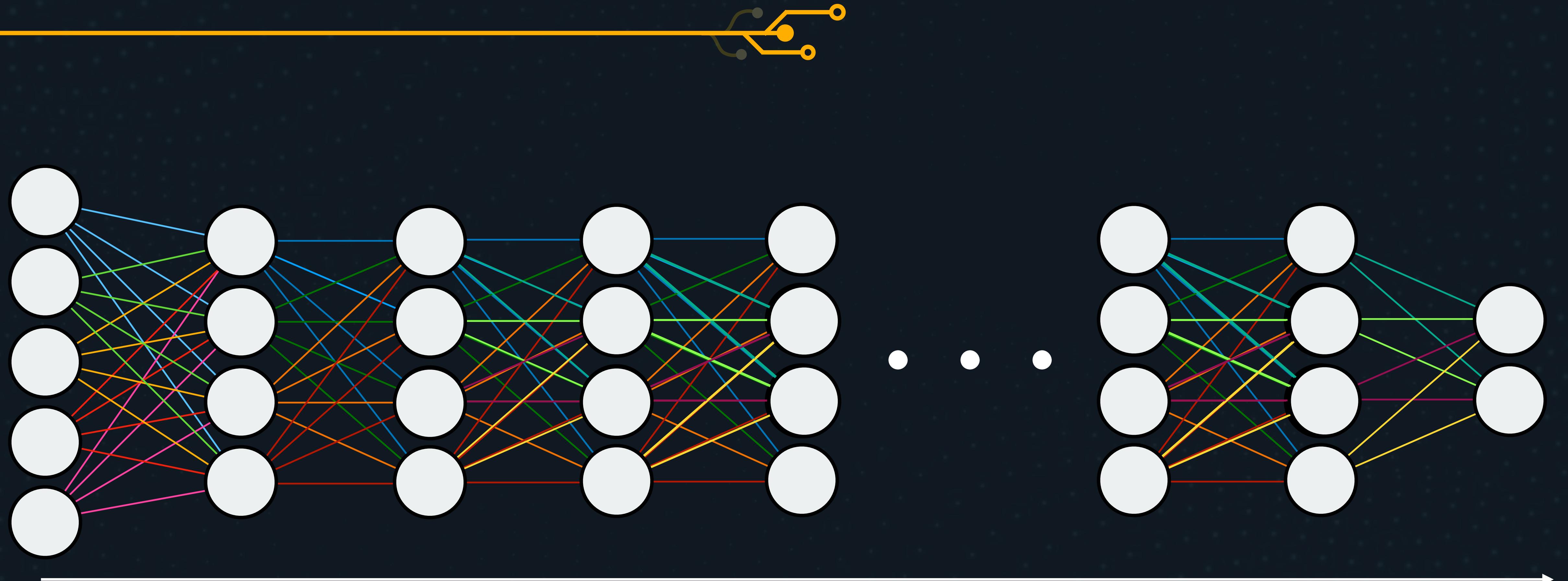
$$f(x \cdot w + b)$$



# Allenamento (Training)

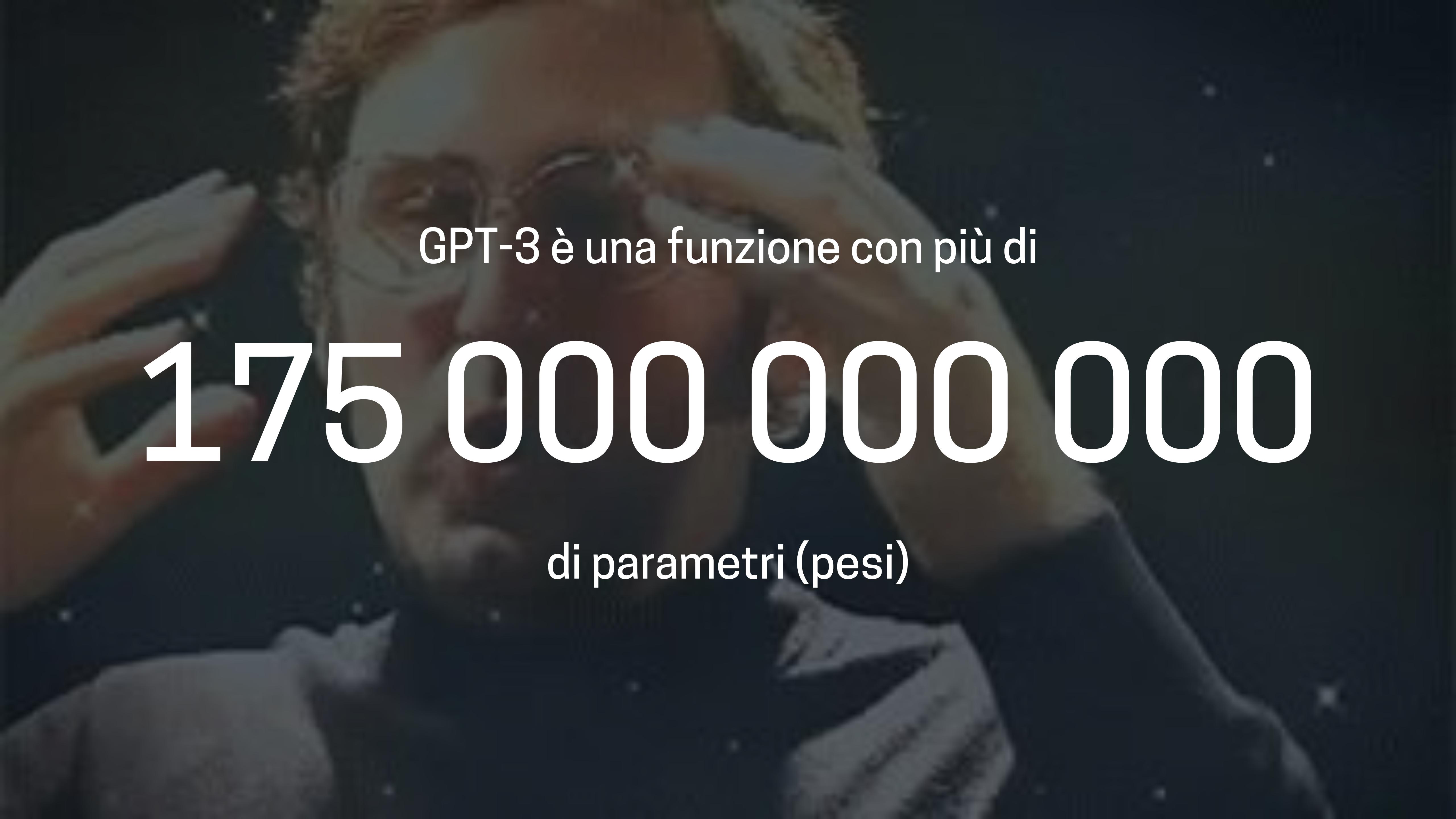


# Deep Learning



Reti molto profonde (Deep)

Ma quanto Deep?



GPT-3 è una funzione con più di  
**175 000 000 000**  
di parametri (pesi)

Parte 3

# Quantum Computing



# Richard Feynman (1918-1988)



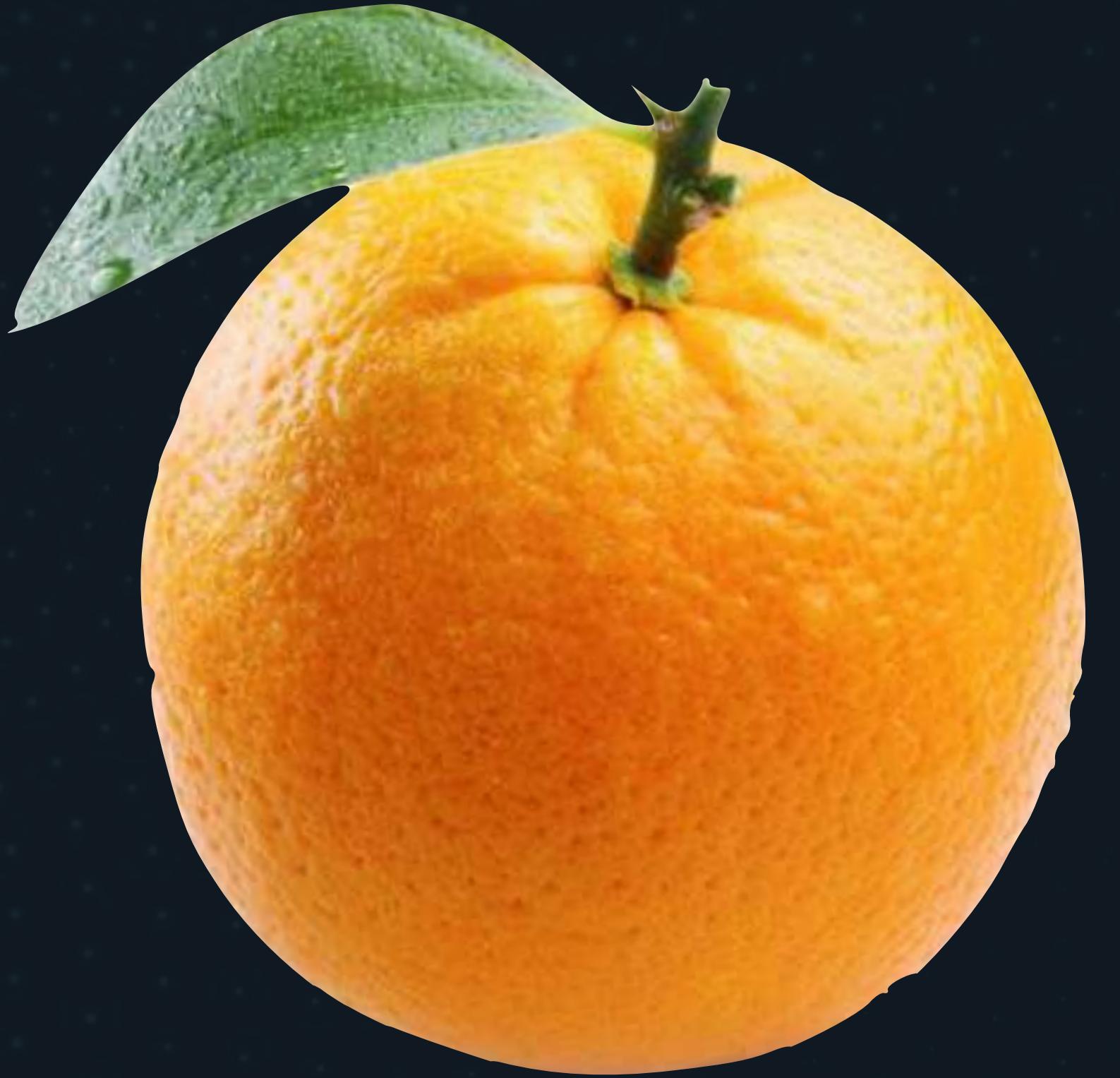
*“La natura non è classica,  
dannazione, e se volessi fare una  
simulazione della Natura, sarebbe  
meglio la facessi quantistica!”*

# Ricetta per una buona computazione quantistica

## Ingredienti:

1. **Sovrapposizione** quanto basta
2. Un spolverata di **Entanglement**
3. **Interferenza** a piacere



An orange with a textured, slightly mottled surface, showing some green at the stem. It is positioned on the left side of the word "bit".

bit

A bright green apple with water droplets on its skin. It has a small brown stem and a single green leaf attached. It is positioned on the right side of the word "bit".

# Qubit



# **Qubit:** unità di calcolo fondamentale di un computer quantistico



## Bit

1

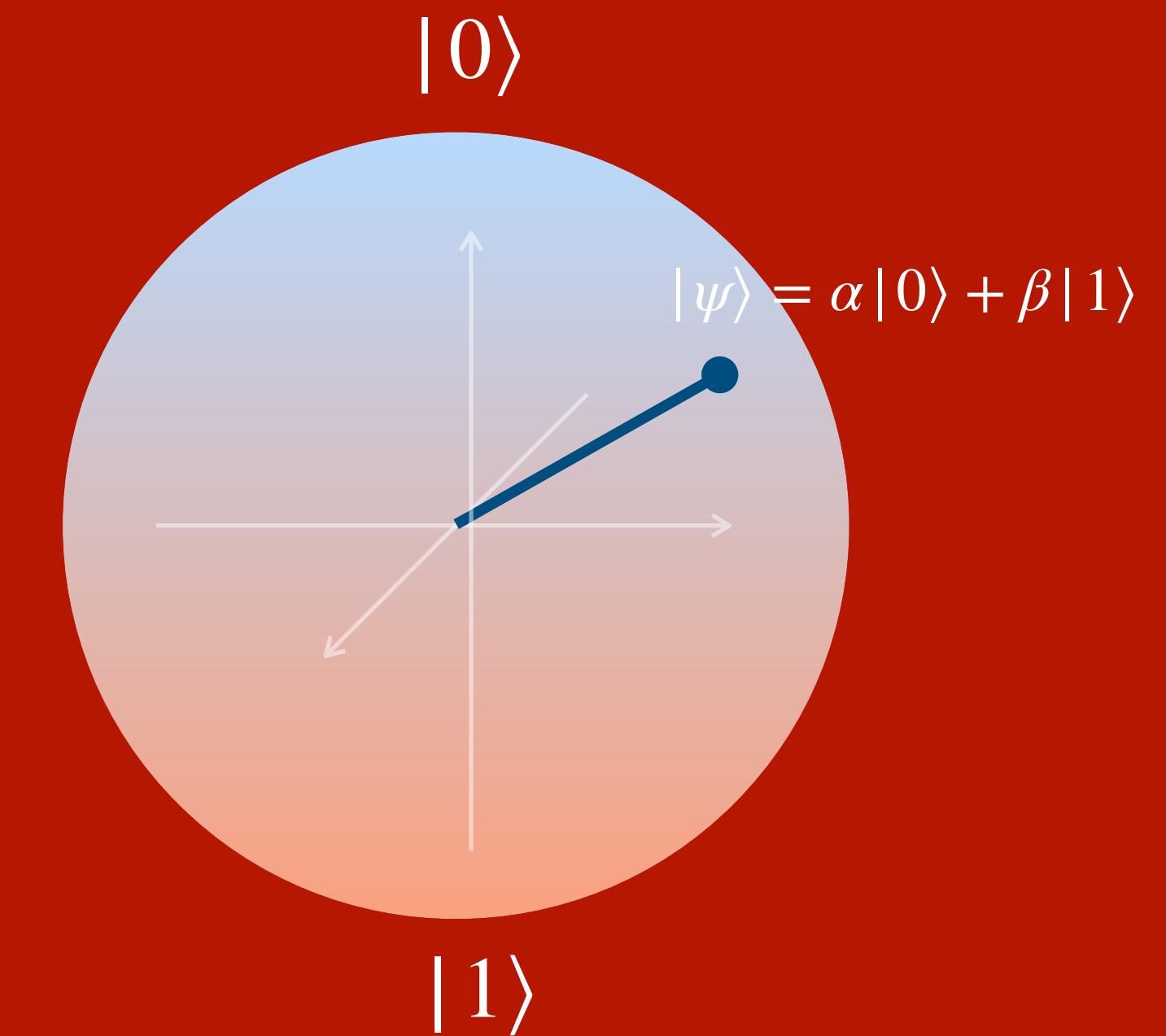
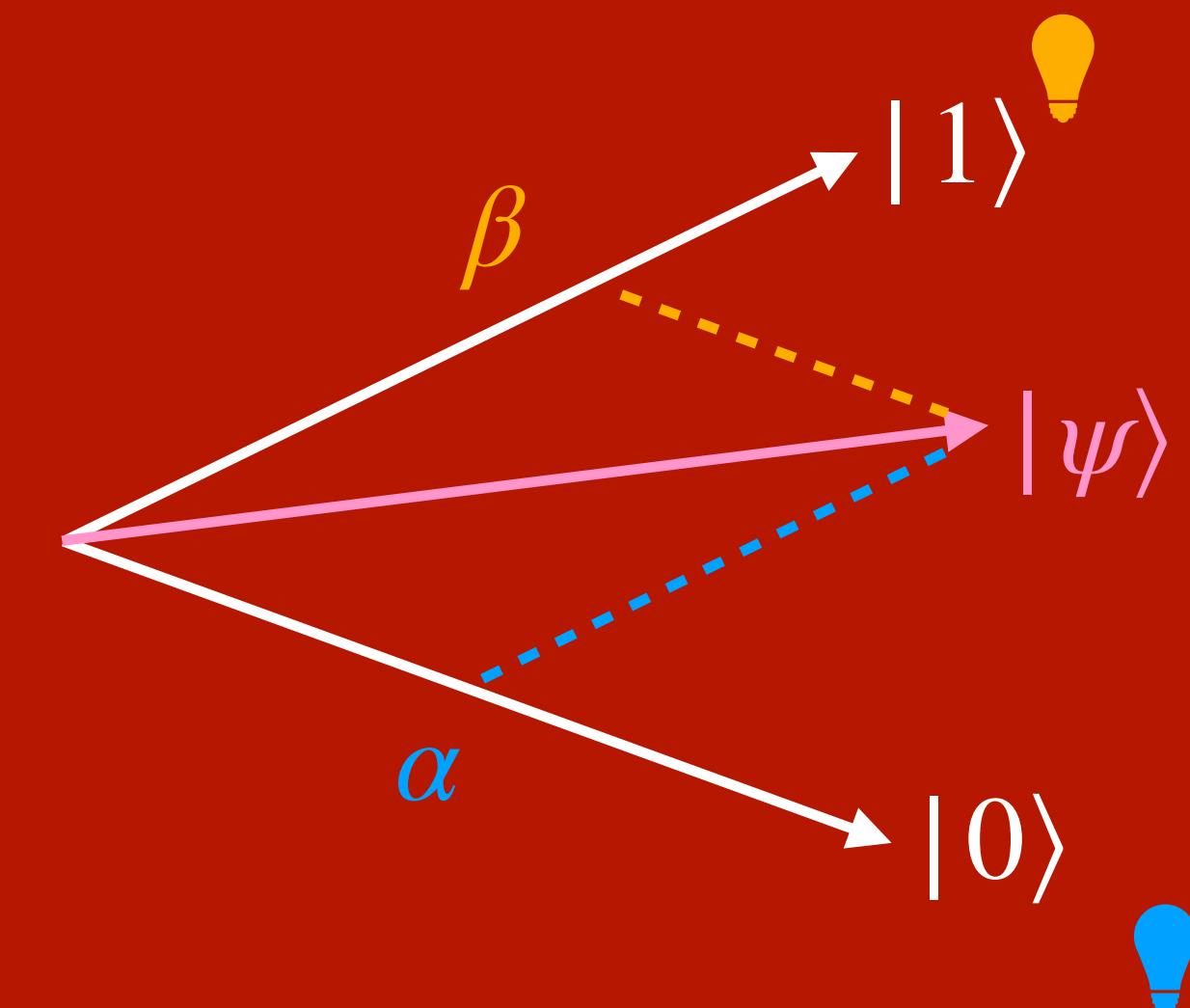
0



Può essere acceso **o** spento

## Quantum Bit (Qubit)

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$



Può essere acceso **e** spento  
allo stesso tempo

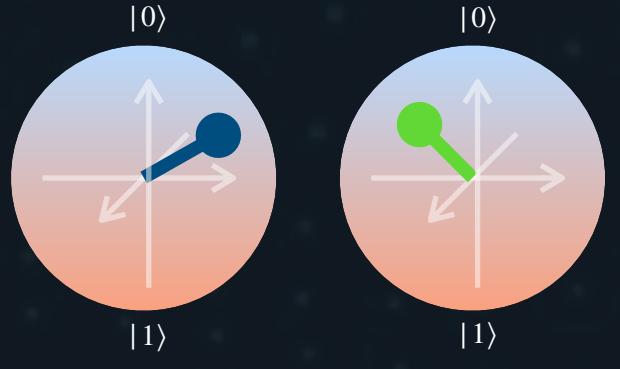
# Più qubits insieme...

1 qubit



$$|\Psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$$

2 qubit



$$|\Psi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle$$

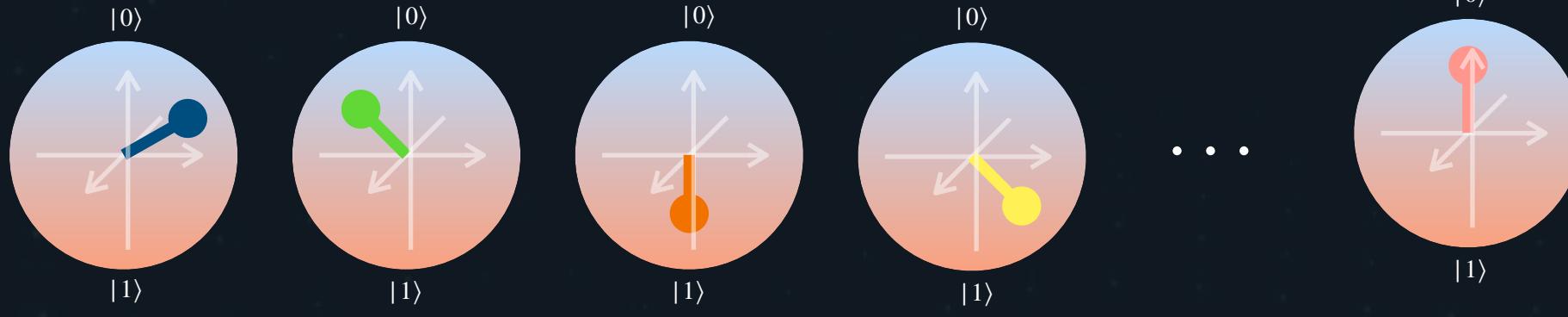
3 qubit



$$|\Psi\rangle = \alpha_0|000\rangle + \alpha_1|001\rangle + \alpha_2|010\rangle + \alpha_3|011\rangle + \\ + \alpha_4|100\rangle + \alpha_5|101\rangle + \alpha_6|110\rangle + \alpha_7|111\rangle$$

⋮

N qubit



$$|\Psi\rangle = \alpha_0|00\dots 0\rangle + \alpha_1|0\dots 01\rangle + \alpha_2|0\dots 10\rangle + \dots \\ + \alpha_{2^N-1}|111\dots 1\rangle$$

$2^N$

Quantità di informazione aumenta esponenzialmente col numero di qubits!

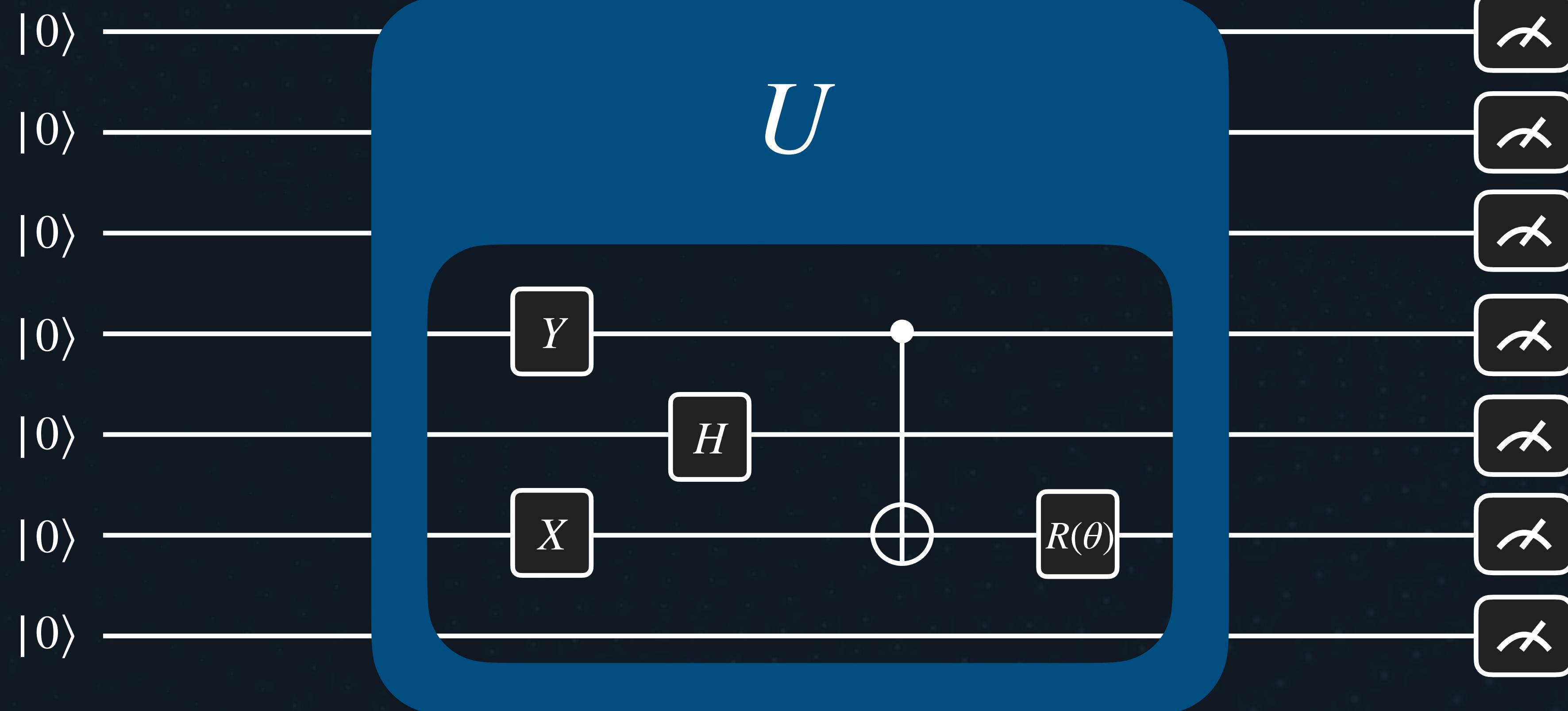
# Schema generale di una computazione quantistica



# 1. Preparazione dei qubits

# 2. Modifichiamo i qubits con dei quantum gates

# 3. Leggiamo il risultato



# 1. Algoritmo di ricerca di Grover



Cercare un **oggetto** tra  $N$  elementi disordinati



**Classicamente** in media devi provare tutte le chiavi per trovare quella giusta

$\sim N$  operazioni

**Quantisticamente**, puoi trovare la chiave giusta con sole

$\sim \sqrt{N}$  operazioni!

... rimane un vantaggio polinomiale.  
È un problema in **P**

Classico **P**

Quantistico **P** (ma più facile)

## 2. Algoritmo di fattorizzazione di Shor



$N = 25195908475657893494027183240048398571429282126204032027777137836043662020707595$   
 $55626401852588078440691829064124951508218929855914917618450280848912007284499268$   
 $73928072877767359714183472702618963750149718246911650776133798590957000973304597$   
 $48808428401797429100642458691817195118746121515172654632282216869987549182422433$   
 $63725908514186546204357679842338718477444792073993423658482382428119816381501067$   
 $48104516603773060562016196762561338441436038339044149526344321901146575444541784$   
 $24020924616515723350778707749817125772467962926386356373289912154831438167899885$   
 $040445364023527381951378636564391212010397122822120720357$

$N = a \cdot b \cdot c \cdot \dots$  con  $a, b, c, \dots$  primi

Classico **NP**

$\sim \exp(d^{1/3})$

Quantistico **P**

$\sim d^3$

( $d$  è il numero di cifre di  $N$ )

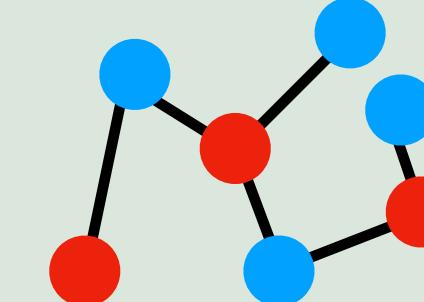
# 3. Simulazione sistemi fisici



$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle$$

Perfetti per un computer quantistico!

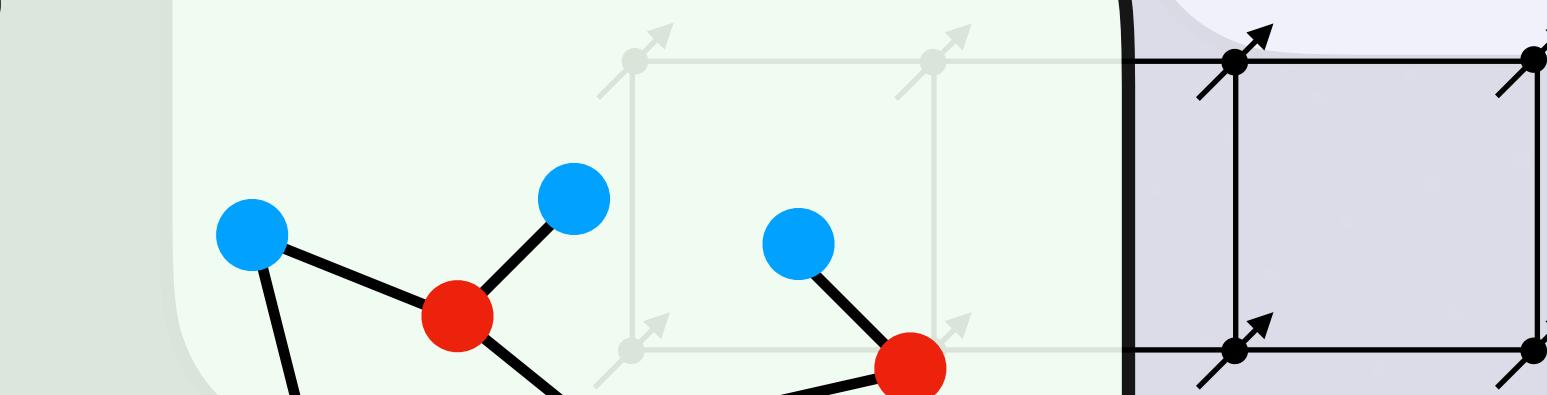
## Modello Fisico



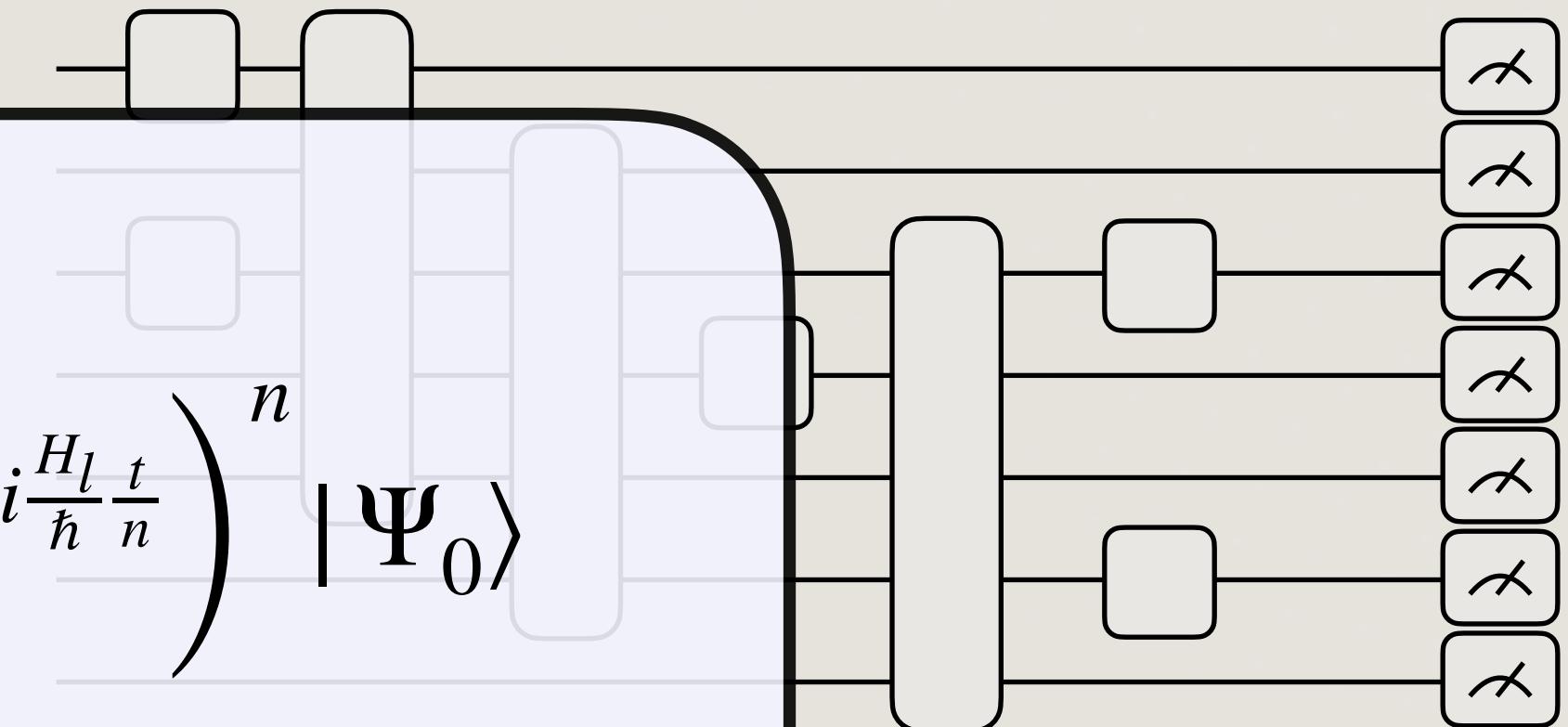
$$|\Psi_0\rangle \xrightarrow{e^{-iHt/\hbar}} |\Psi(t)\rangle$$

## Hamiltoniana di Pauli

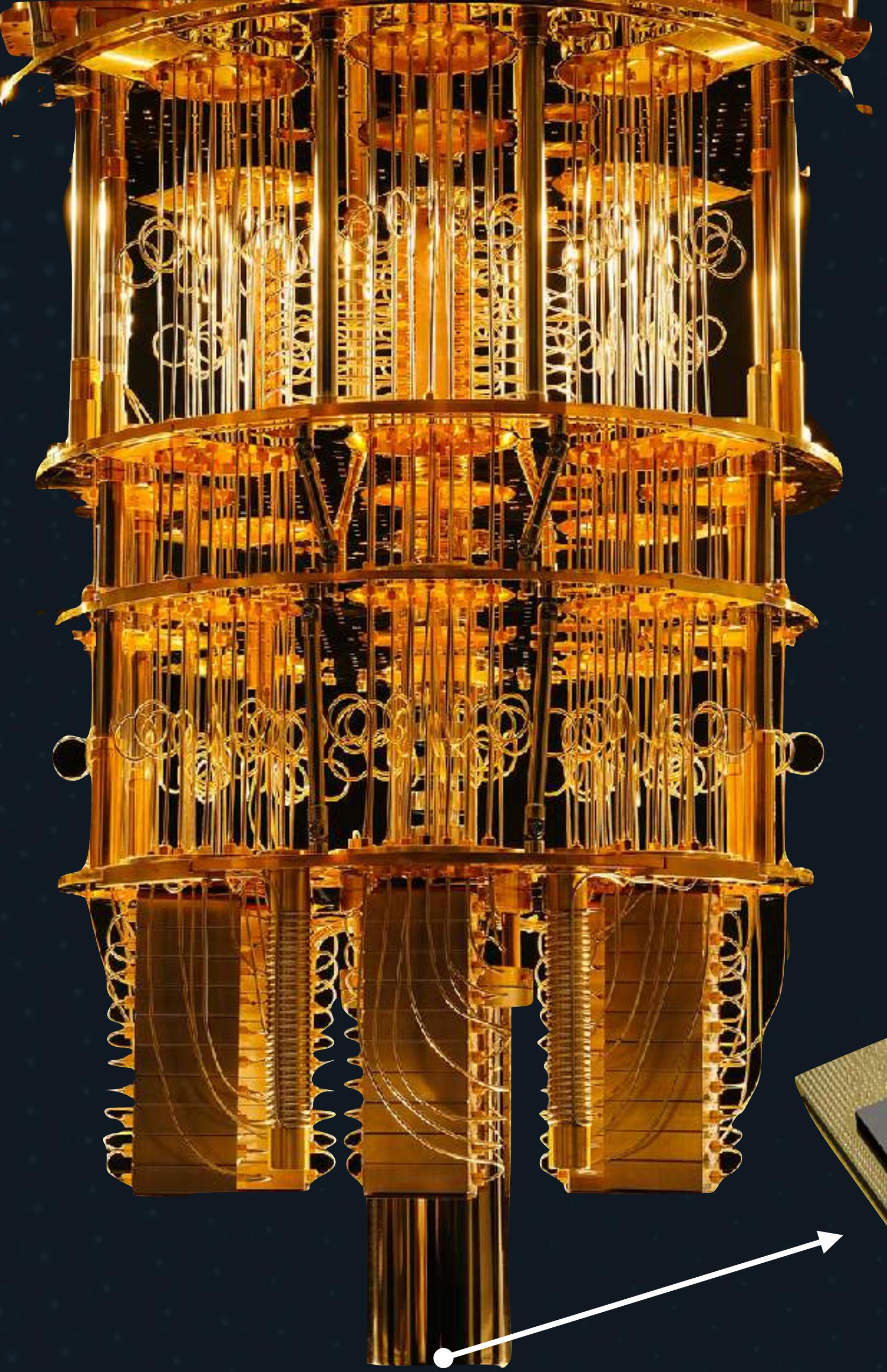
$$|\Psi(t)\rangle \approx \left( \prod_l e^{-i\frac{H_l}{\hbar} \frac{t}{n}} \right)^n |\Psi_0\rangle$$



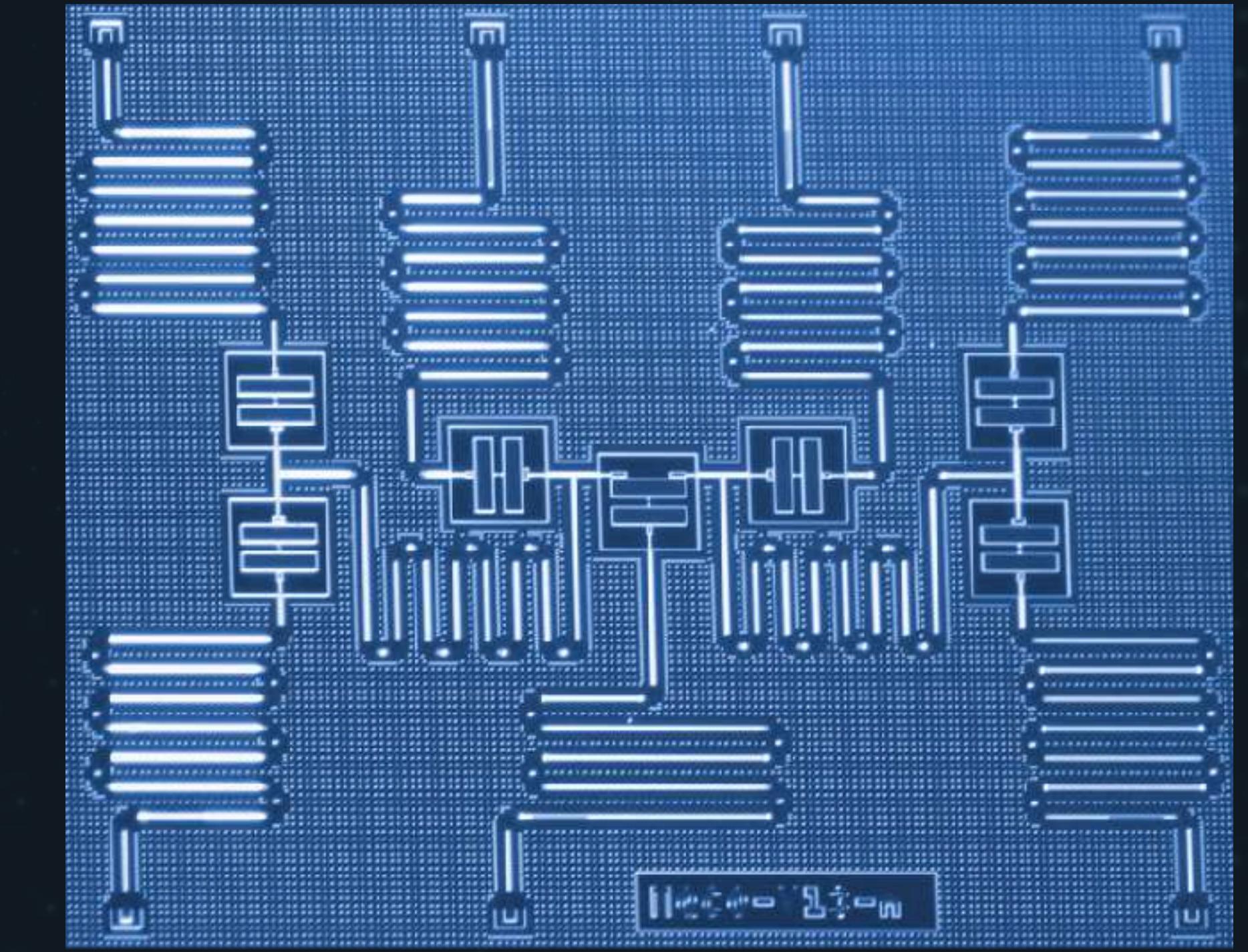
## Circuito Quantistico



# Un Quantum Computer (o meglio, un frigorifero)

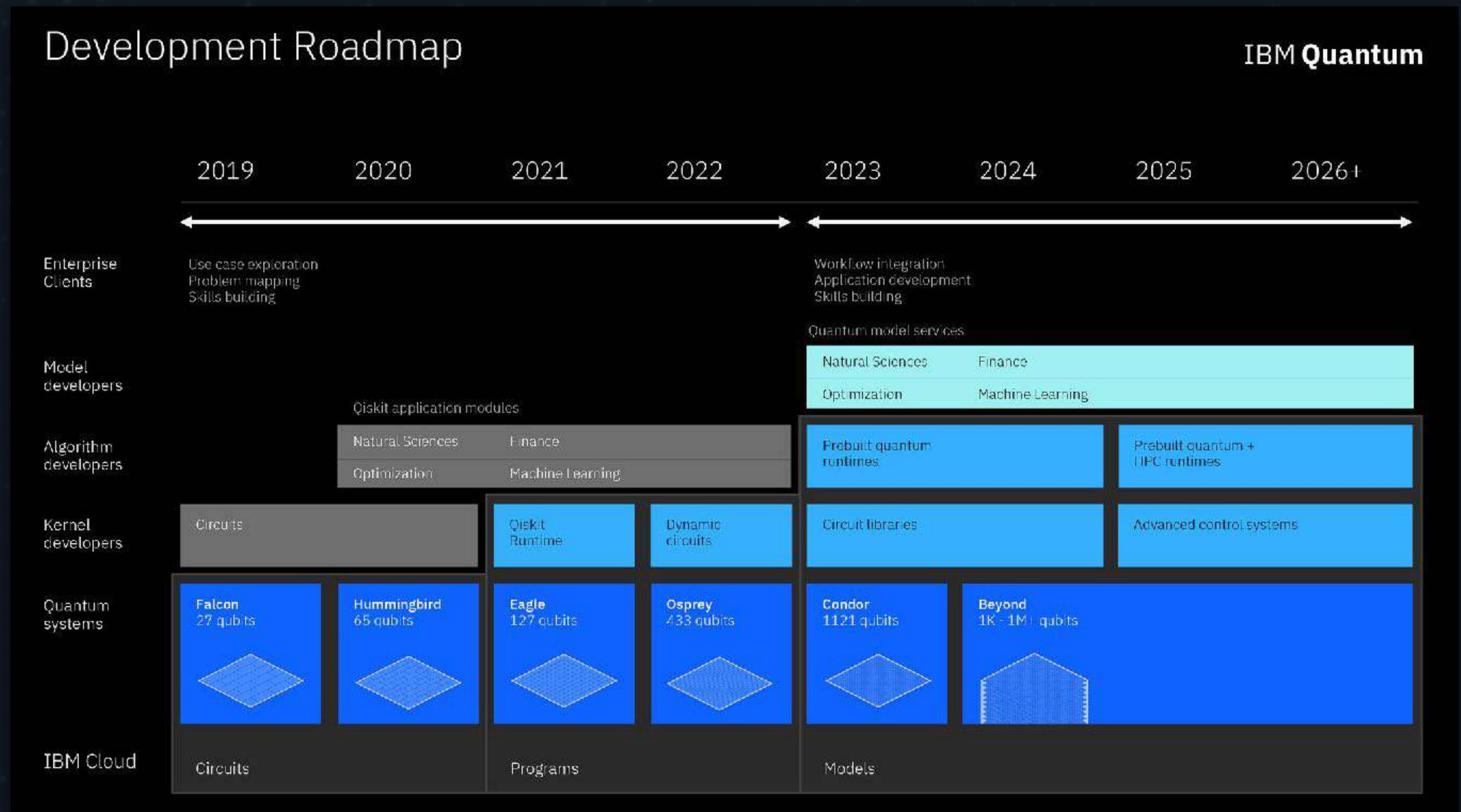


Temperatura di  
 $-270^{\circ}\text{C}$



Varie tecnologie per realizzarli:  
Circuiti Superconduttori, Fotonica, Atomi Neutri, Trappole a Ioni, ...

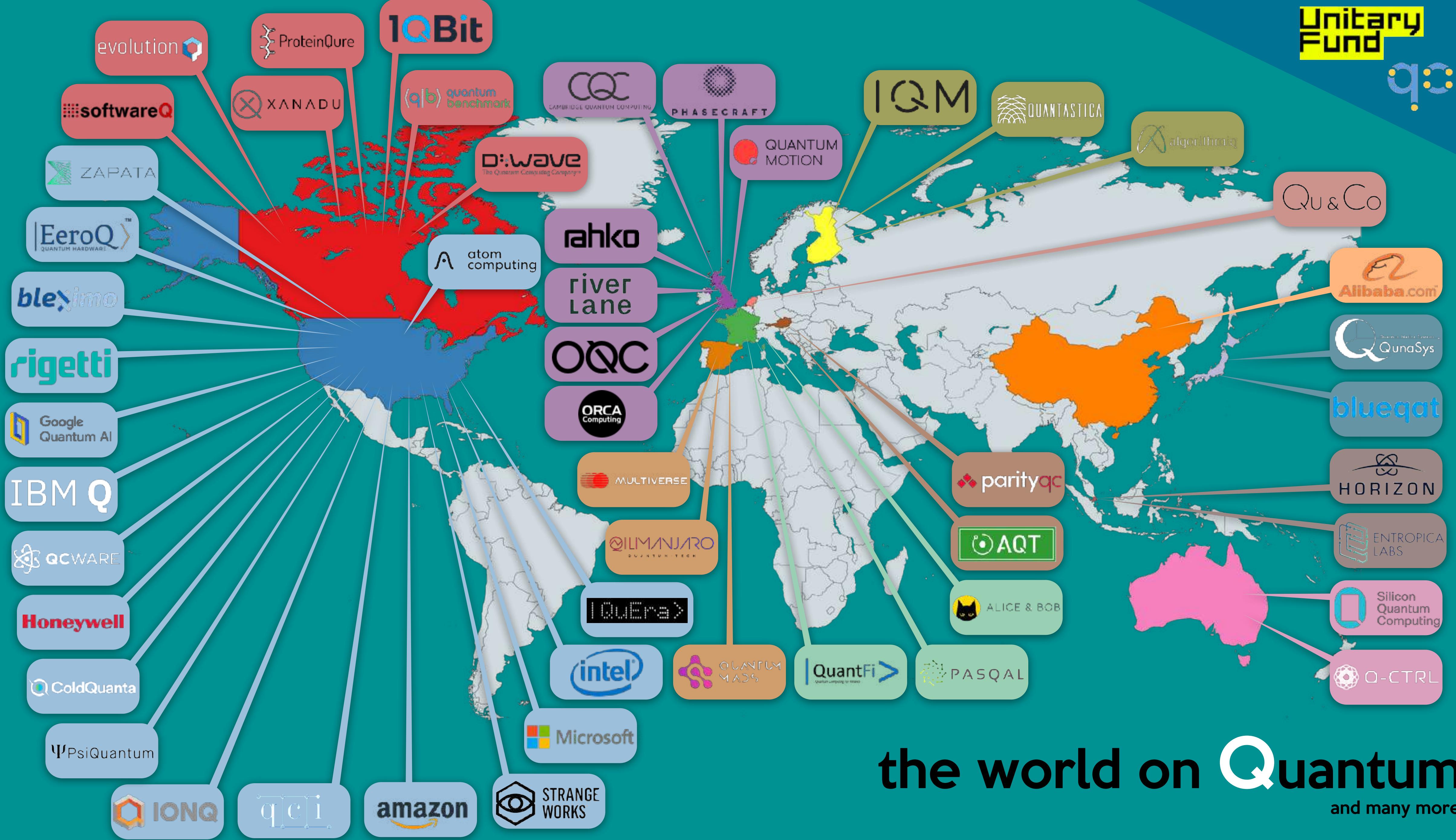
# La strada è ancora lunga...



Per rendere Quantum Computer utili, servono:

- Molti più qubits
- Qubits “meno rumorosi”
- Correzione degli errori
- Nuovi algoritmi

the world on **Quantum.**  
and many more...



# Potete programmare anche voi un computer quantistico!



## Cirq



## Qiskit



## PENNY LANE



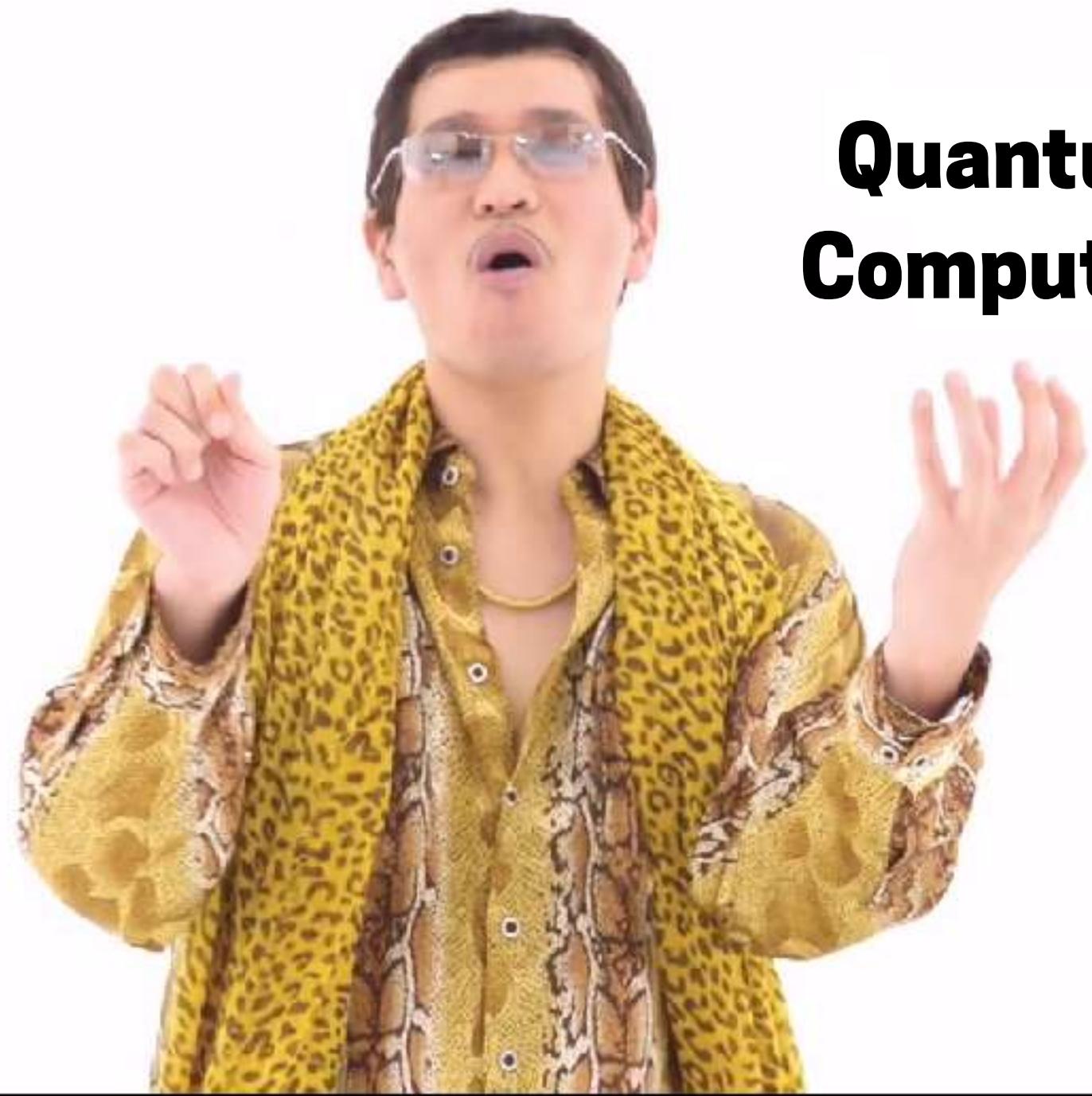
## TensorFlow Quantum

e molti altri!

**Machine  
Learning**



**Quantum  
Computing**

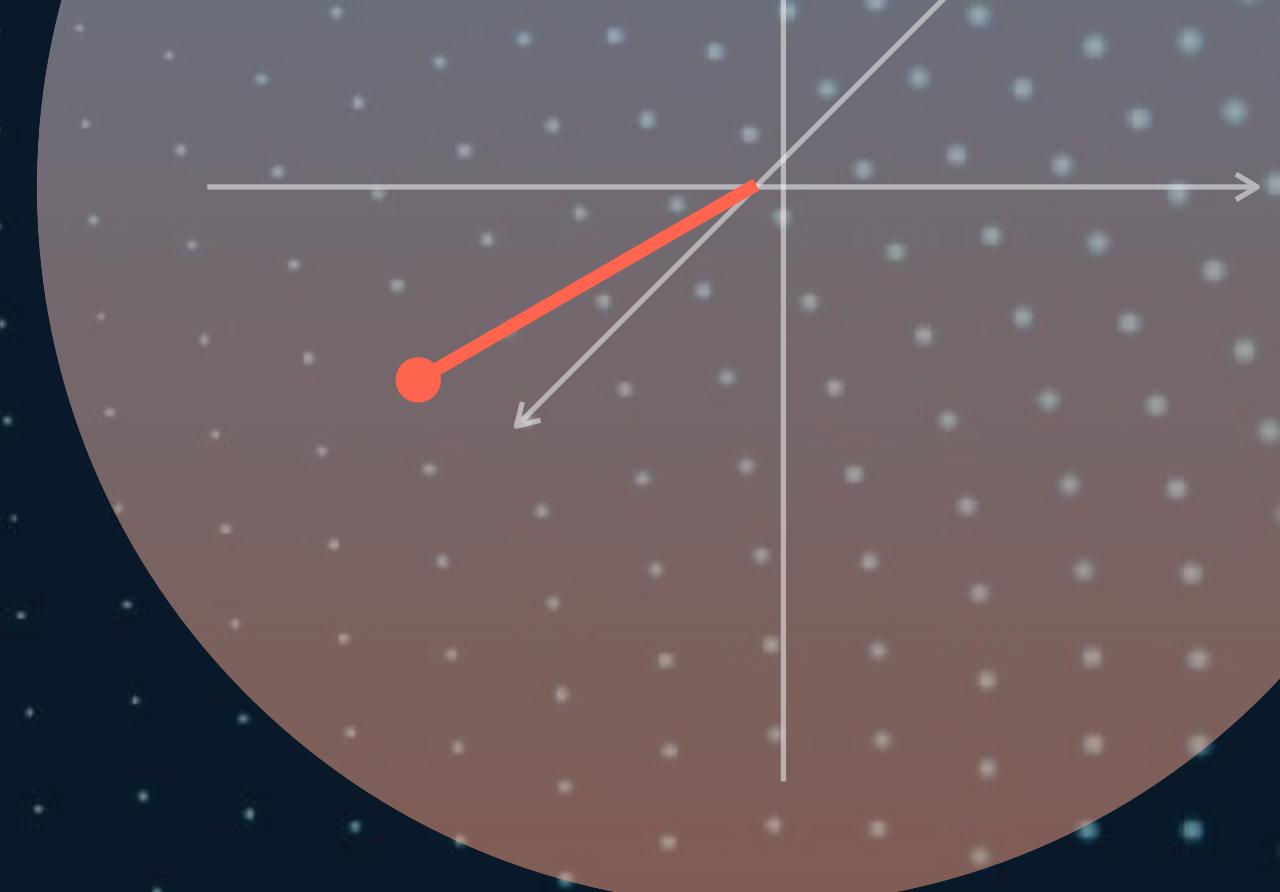


**mmh!**



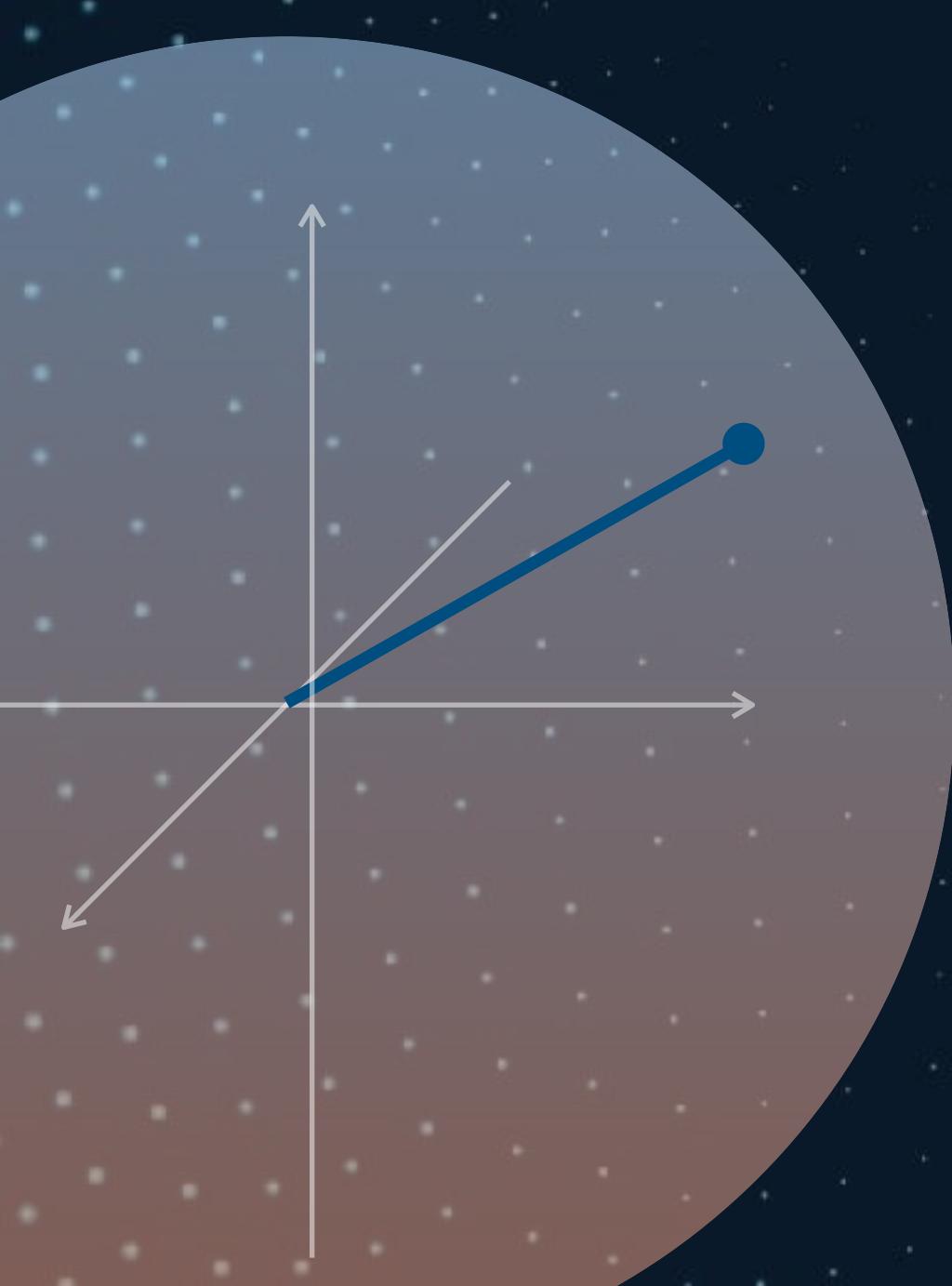
**Quantum Machine Learning!?**





...the end

# Grazie per l'attenzione!



**non esitate a contattarmi se avete domande:**



[stefano.mangini01@universitadipavia.it](mailto:stefano.mangini01@universitadipavia.it)



@stfnmangini



Stefano Mangini

# Risorse per chi è interessato

---



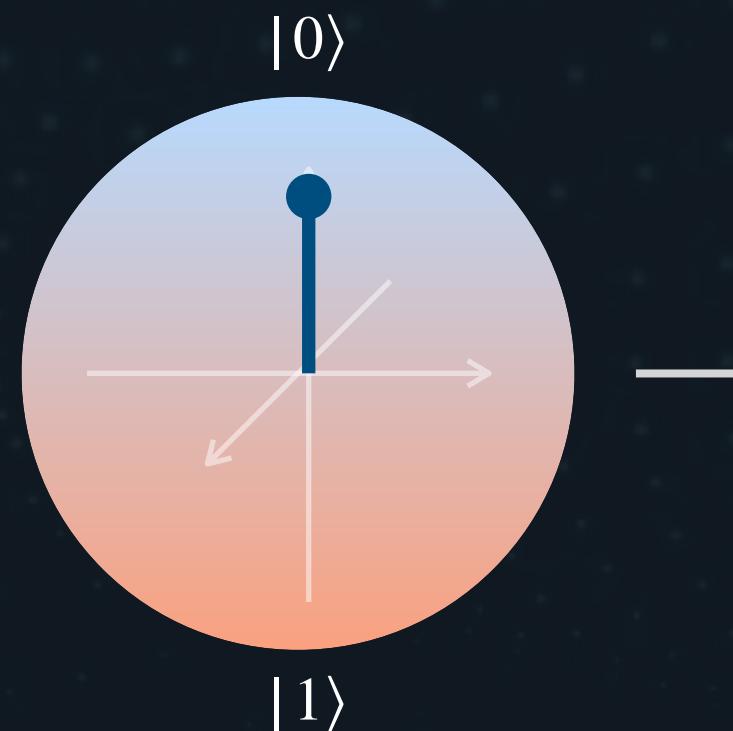
## Libri

- Quantum Computation and Quantum Information, Nielsen & Chuang
- Quantum Computing Since Democritus, Scott Aaronson
- Un'occhiata alle carte di Dio, Gian Carlo Ghirardi

## Risorse online

- Quantum Open Source Foundation (qosf), <https://www.qosf.org/>
- Qiskit Textbook, <https://qiskit.org/textbook/preface.html>
- Pennylane's demos, <https://pennylane.ai/qml/demonstrations.html>
- Quantum Algorithms for data analysis, <https://quantumalgorithms.org>
- 3Blue1Brown (per Algebra Lineare), [https://www.youtube.com/watch?v=fNk\\_zzaMoSs](https://www.youtube.com/watch?v=fNk_zzaMoSs)
- TwoMinutePapers (per IA), <https://www.youtube.com/user/keeroyz>
- ...

# Operazioni sui qubits... *Quantum Gates*

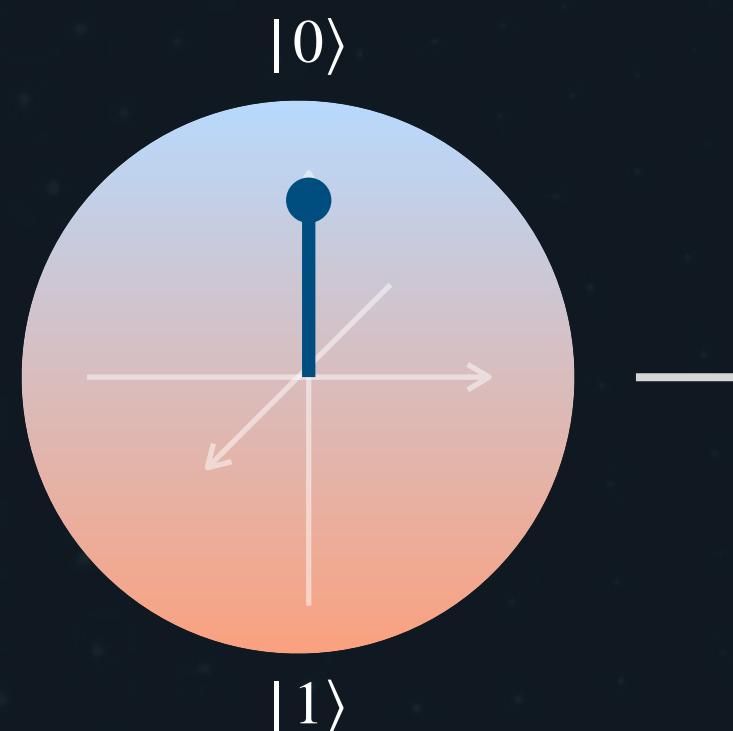


Not  
 $X$

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$


$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle$$



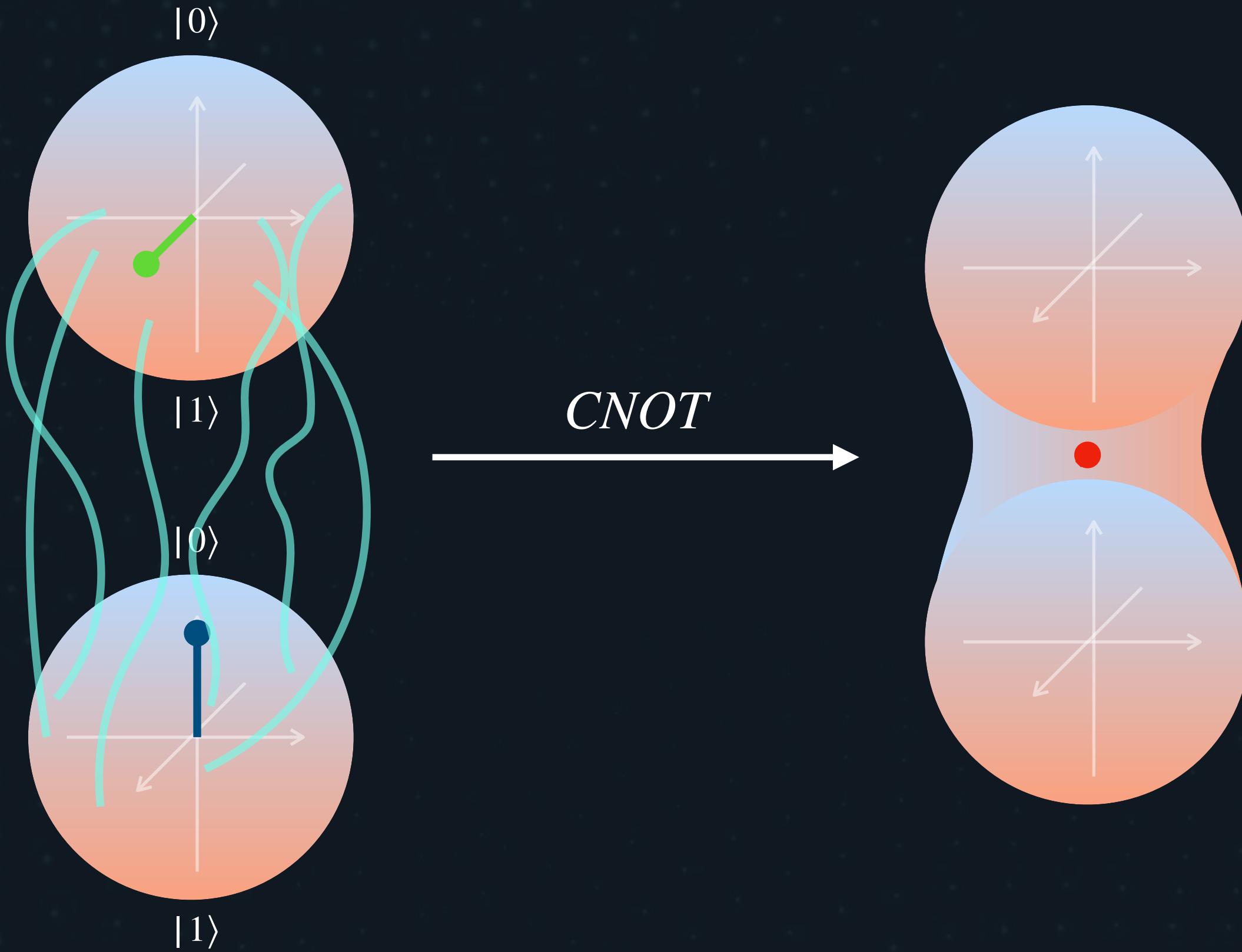
Hadamard  
 $H$

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$$


$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$= \frac{1}{\sqrt{2}} \left( \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right) = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) = |+\rangle$$

Sovrapposizione!

# Operazioni a due qubits...*Entanglement!*



$$|\Psi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle$$

$$\alpha_0 = \alpha_3 = \frac{1}{\sqrt{2}} \quad \alpha_1 = \alpha_2 = 0$$

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Stato di un qubit è indissolubilmente legato all'altro!

Combinando gates ad 1 e 2 qubits, si può fare tutto!