

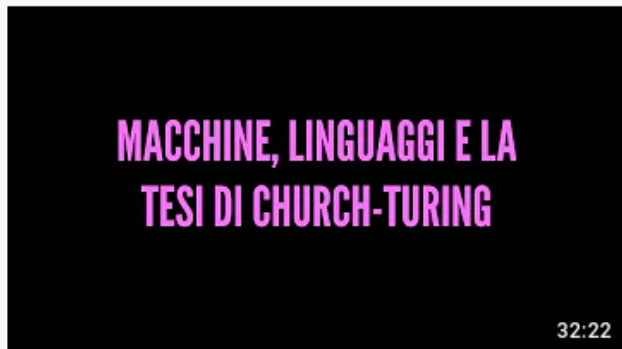
LA MACCHINA DI TURING

TABLE OF CONTENTS

- Riassunto
- Alan M. Turing
- La Macchina di Turing
 - Definizione Matematica
 - Esempio #1: somma unaria
 - Esempio #2: somma binaria
- La Macchina Universale di Turing

RIASSUNTO

Nella precedente lezione avevamo introdotto le connessioni tra i **linguaggi di programmazione** e le **macchine** (modelli di calcolo) in grado di leggere ed eseguire tali linguaggi.



Macchine, Linguaggi e la Tesi di Church-Turing

83 visualizzazioni • 6 giorni fa



Leonardo Tamiano

In questo video introduco il concetto di modello di calcolo descrivendo in particolare le relazioni che sussistono tra macchine e ...

Nuovo

[Link al video](#)

Avevamo poi chiuso il video menzionando il fatto che di modelli di calcolo nel corso degli anni ne sono stati definiti tanti, tra cui:

- Macchine di Turing.
- Funzioni ricorsive.
- Lambda calcolo.
- Le macchine a registri elementari.
- I comuni linguaggi di programmazione.

Eppure, la **tesi di Church-Turing** menziona solamente le
macchine di Turing:

**È calcolabile tutto (e solo) ciò che può essere calcolato
tramite una macchina di Turing.**

In questa lezione andiamo a vedere che cos'è esattamente una macchina di Turing, e perché le macchine di Turing sono un ottimo modello matematico per rappresentare il concetto intuitivo di **calcolabilità**.

ALAN M. TURING

Prima di entrare nei dettagli tecnici, è doveroso fare qualche nota sull'autore di questo modello di calcolo, **Alan M. Turing**, che oggi è considerato uno dei padri fondatori dell'informatica teorica e dell'intelligenza artificiale.

Alan M. Turing nasce nel 1912 in Inghilterra e muore dopo 42 anni, nel 1954, in seguito ad un morso dato ad una mela intinta col cianuro.

Sono tanti e vari i lavori svolti da Turing nel breve tempo che ha avuto a disposizione. Tra questi spiccano di importanza e rilevanza i seguenti:

- Ha definito il concetto di **macchina di Turing**. (1936)
- Ha dato notevoli contributi alla decifrazione delle comunicazioni tedesche cifrate con la macchina **Enigma**. (1939-1945)
- Ha scritto i primi articoli sull'intelligenza artificiale, introducendo il famoso **test di Turing** (1950).

La vita e le opere di Turing sono troppo complesse per essere riassunte in pochi minuti.

C'è solo una cosa che nella nostra ignoranza possiamo dire riguardo ad una persona come Turing:

Ha dedicato la sua ingegnosità e la sua curiosità per proteggere le persone a lui vicine, e queste ultime lo hanno condannato per la sua sessualità.

LA MACCHINA DI TURING

La strada che ha portato Turing a definire il modello di calcolo oramai noto con il suo nome è, come tante strade umane, lunga, tortuosa, e affascinante.

Per capire appieno il motivo per cui Turing si è imbattuto nell'idea della macchina di Turing si dovrebbe iniziare dall'alba dell'umanità.

L'obiettivo finale della macchina di Turing infatti è **lo studio dei processi automatici**. Detto altrimenti,
il calcolo automatico

Questa strada in particolare è strettamente legata alla **crisi dei fondamenti della matematica** avvenuta agli inizi del 1900.

Data la complessità dell'argomento, rimando la trattazione ad un futuro video.

L'unica cosa importante da dire è che Turing stava cercando di risolvere un problema molto importante nel campo della **logica matematica**.

Il famoso paper del 1936 che introduce la macchina di Turing è noto con il nome di

230

A. M. TURING

[Nov. 12,

ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. TURING.

[Received 28 May, 1936.—Read 12 November, 1936.]

[Link al paper](#)

Prima ancora di entrare nei dettagli matematici, Turing pone il lettore di fronte ad una situazione ben nota:

quella di una persona in procinto di eseguire un calcolo

Cosa succede, ad esempio, quando una persona sta cercando di calcolare la somma di due numeri?

A handwritten addition problem on a blue grid background. The first row shows the numbers 1337 with a plus sign to its right. Above the second 3 is a small '+1'. The second row shows the numbers 44. A horizontal line is drawn below the second row. The third row shows the result 1381.

$$\begin{array}{r} 1337 + \\ \quad 44 \\ \hline 1381 \end{array}$$

Osservazione: Un tempo la parola **computer** non si riferiva alle macchine, ma a degli esseri umani (tipicamente donne) il cui lavoro era quello di effettuare dei calcoli matematici molto difficili.

In altre parole,

Computer è ciò che calcola

Nel calcolo, Turing cattura i seguenti aspetti (1/3):

La persona, per lavorare, utilizza un pezzo di **carta**.

Su questa carta la persona può scrivere vari **simboli**.

In ogni momento del calcolo, la persona è in grado di leggere solo una **porzione finita** della carta su cui sta lavorando.

Nel calcolo, Turing cattura i seguenti aspetti (2/3):

In ogni momento del calcolo, la persona si troverà in un proprio **stato interno**.

Lo stato interno modifica come la persona reagisce alla visione di determinati simboli.

Nel calcolo, Turing cattura i seguenti aspetti (3/3):

A seconda di ciò che vede, e a seconda del proprio stato interno, la persona può decidere di:

1. Cambiare il proprio stato interno.
2. Scrivere un nuovo simbolo sul pezzo di carta, eventualmente sostituendo un simbolo precedentemente scritto.
3. Cambiare la regione della carta su cui sta ponendo l'attenzione.

Notiamo che tutti questi aspetti sono puramente **meccanici**, nel senso che non fanno utilizzo né di intuizione e né di creatività.

Ed è proprio nel seguente passaggio che si trova la novità geniale introdotta da Turing:

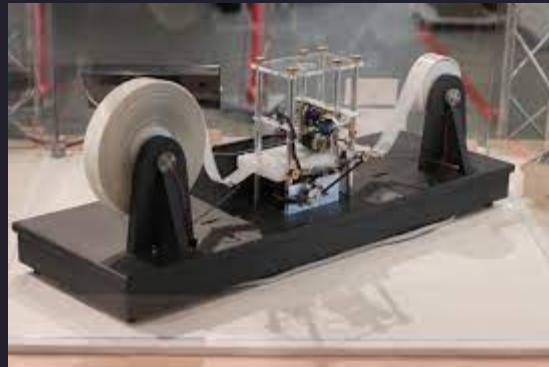
Essendo meccanici, tutti questi passi possono potenzialmente essere effettuati da una macchina.

La carta è sostituita con un nastro diviso in celle, in cui in ogni cella può apparire un simbolo.

Su questa carta lavora una macchina con un proprio stato interno e una testina in grado di leggere, in ogni momento, un solo simbolo dal nastro.

A seconda dello stato interno e del simbolo letto, la macchina può decidere di cambiare stato interno, di scrivere un nuovo simbolo sul nastro, ed eventualmente di spostare la testina di un cella, a destra o a sinistra.

In questo modo, Turing sostituisce l'operatore umano con una macchina potenzialmente costruibile, facendo nascere a tutti gli effetti l'informatica teorica, intesa come lo **studio teorico dei processi di calcolo automatici**.



Source

DEFINIZIONE MATEMATICA

Entriamo adesso nei dettagli matematici del modello.

Una macchina di Turing è definita come una **sestupla**

$$\langle \Sigma, \blacksquare, Q, q_0, Q_F, P \rangle$$

- Σ è un **alfabeto di simboli**.
- \blacksquare è il carattere **blank**.
- Q è un insieme di **stati**.
- $q_0 \in Q$ è lo **stato iniziale**.
- $Q_F \subseteq Q$ è l'insieme degli **stati finali**.
- P è la **funzione di transizione**

La parte più importante di una macchina di turing è la
funzione di transizione P

$$P : (Q - Q_F) \times (\Sigma \cup \blacksquare) \longrightarrow Q \times (\Sigma \cup \blacksquare) \times \{d, s\}$$

Tale funzione infatti definisce le **regole del
comportamento** della macchina. In altre parole,
rappresenta il programma (software) implementato
dalla particolare macchina di Turing.

Ciascuna regola in P è formata da 5 componenti:

$$(q, s) \longrightarrow (q', s', m)$$

Se la macchina si trova nello stato q e legge il simbolo s dal nastro, allora sovrascrive tale simbolo con il simbolo s' , entra nel nuovo stato interno q' e fa il movimento descritto da m .

ESEMPIO #1: SOMMA UNARIA

Per capire meglio il funzionamento di queste macchine
ho deciso di implementare un simulatore di macchine di
Turing scritto in **javascript**.

Simulatore

A seguire, in ordine: la tabella che contiene le istruzioni che governano il **comportamento** della macchina di Turing che si vuole simulare; il nastro contenente l'**input** della macchina; la **posizione della testina**; il pannello di controllo e il pulsante di avvio.

	Stato corrente	Simbolo letto →	Nuovo stato	Simbolo scritto	Movimento
<input data-bbox="176 544 234 604" type="button" value="+"/>	<input data-bbox="284 544 528 604" type="text"/>	<input data-bbox="580 544 825 604" type="text"/>	<input data-bbox="918 544 1162 604" type="text"/>	<input data-bbox="1218 544 1463 604" type="text"/>	<input data-bbox="1524 544 1769 604" type="text" value="F"/>

↓

Stato Interno:

Velocità (ms):

Passi eseguiti:

Calcola!

[Link al simulatore](#)

Ad esempio, consideriamo il problema di sommare due numeri scritti in **unario**, ovvero utilizzando un solo simbolo...

ESEMPIO #2: SOMMA BINARIA

Un problema assai più difficile da risolvere utilizzando una macchina di turing è il problema di sommare due numeri scritti in **binario**...

La difficoltà nel risolvere tale problema è conseguenza diretta del fatto che la macchina di Turing è un modello estremamente semplice di calcolo:

Potendo fare solamente poche cose, bisogna spezzare operazioni più complesse, come la somma di due numeri scritti in binario, in tantissime istruzioni più semplici.

Per questa ragione quando si scrivono degli algoritmi non si programma a livello delle **quintuple**, ma si utilizzano linguaggi più astratti.

Detto questo, tutto ciò che scriviamo in un qualsiasi linguaggio di programmazione può essere, in linea teorica, tradotto in delle quintuple di una macchina di Turing.

LA MACCHINA UNIVERSALE DI TURING

Osserviamo a questo punto che
una singola macchina di Turing permette di risolvere
un particolare problema computazionale

$$T_1 \longrightarrow P_1$$

$$T_2 \longrightarrow P_2$$

$$T_3 \longrightarrow P_2$$

L'obiettivo di Turing nel suo paper del 1936 però non era la semplice definizione di una macchina di Turing.

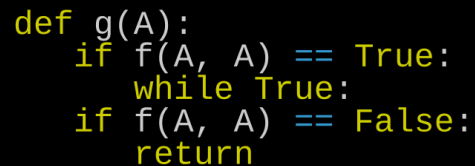
Lui era interessato a dimostrare un risultato molto più generale:

**Esistono problemi computazionali che non possono
essere risolti in modi automatici**

In un video precedente che ho fatto sul canale ho trattato proprio questo problema con tutti i dettagli matematici del caso.

Algoritmi e Strutture Dati

02 - Il Problema della Fermata

A code window with a white border and three colored window control buttons (red, yellow, green) in the top-left corner. It contains Python code for a function g(A) that checks if f(A, A) is True or False using a while loop.

```
def g(A):  
    if f(A, A) == True:  
        while True:  
            if f(A, A) == False:  
                return
```

[Link al video](#)

Per raggiungere il suo obiettivo doveva quindi argomentare che la macchina da lui definita, ovvero la macchina di Turing, era in grado di catturare il concetto intuitivo di **calcolabilità**.

Turing voleva far capire al lettore che

**Se un problema computazionale poteva essere risolto,
allora esisteva una particolare macchina di Turing in
grado di risolvere tale problema**

E, viceversa,

Che se non esisteva una macchina di Turing in grado di risolvere un particolare problema computazionale, allora quel problema non poteva essere risolto in modo automatico

Questa tesi è stata poi formalizzata con il nome di **Tesi di Church-Turing.**

Per aiutare il lettore a capire questa tesi, Turing
introduce la sua idea più bella di sempre:

la macchina universale di Turing

Mentre una macchina di Turing è in grado di risolvere un solo problema computazionale, come la calcolatrice è in grado di risolvere solamente problemi di tipo aritmetico, la macchina universale di Turing è "universale" nel senso che è in grado di risolvere tutti i problemi computazionali che ammettono una soluzione automatica

Per fare questo l'idea geniale dietro alla macchina universale di Turing è la seguente:

Codificare le istruzioni di una qualsiasi macchina di Turing sotto forma di dati, ed interpretare queste istruzioni in modo da simulare il comportamento di quella particolare macchina.

In altre parole,

Dati ed istruzioni sono memorizzati nello stesso modo

In altre parole, la macchina universale U prende in input due cose:

1. Le istruzioni P_T di un'altra macchina di Turing T
2. I dati D_T in input alla macchina T

E, tramite questi dati, è in grado di **simulare** il comportamento di T sui dati D_T .

Una volta costruita la macchina di turing U , per modificarne il comportamento ci basta modificare i dati che passiamo in input alla macchina.

La macchina universale di Turing è uno dei primi esempi di macchina programmabile tramite del software.

