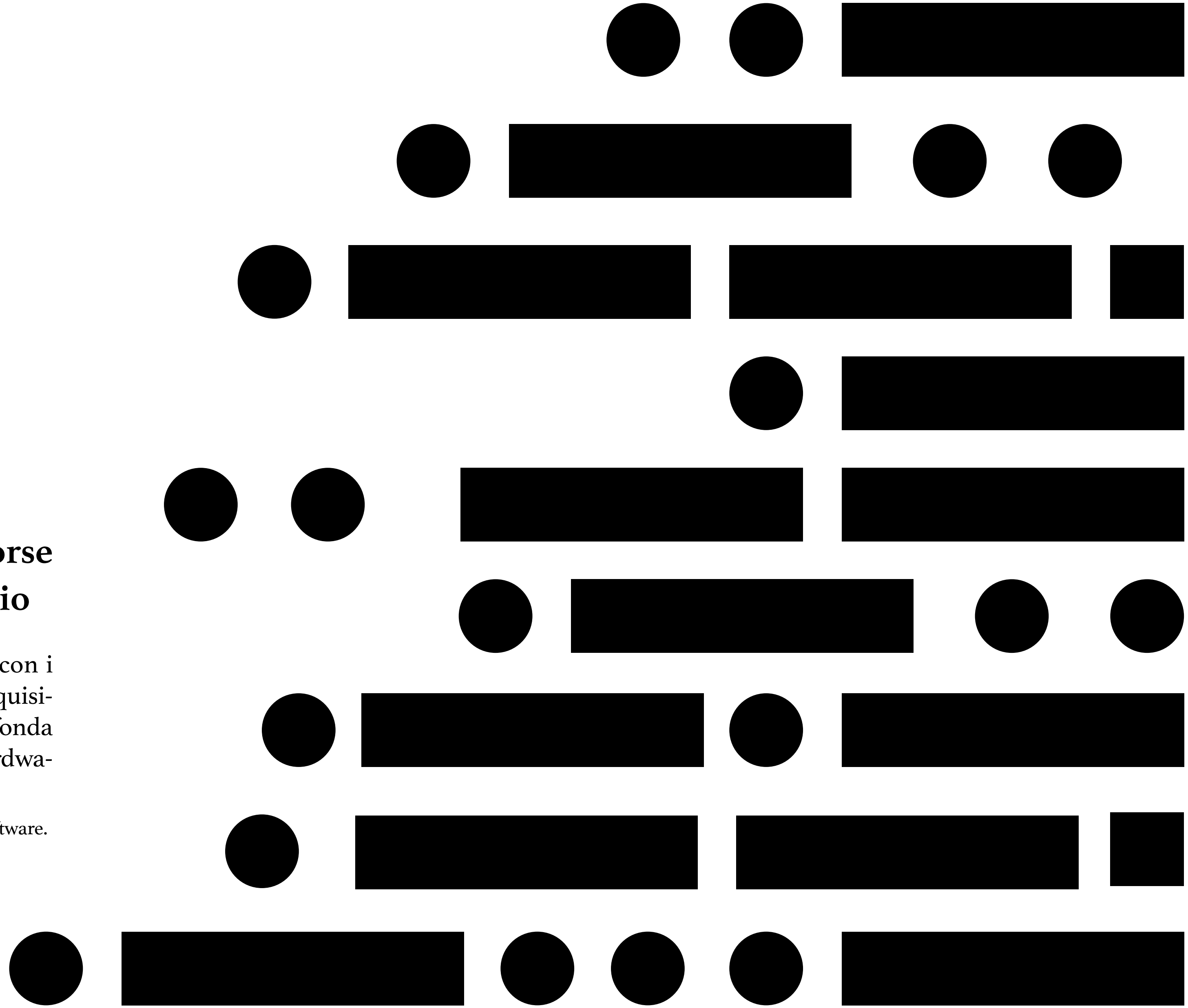


Punti, Linee e Logica: Il Codice Morse come Matrice del Linguaggio Binario

“Anche se il codice Morse non ha nulla a che fare con i computer, avere chiara la natura dei codici è un prerequisito essenziale per raggiungere una comprensione profonda dei linguaggi nascosti e delle strutture interne dell’hardware e del software.”

- Petzold, Charles (2023). Code: Il linguaggio segreto di computer e software.

01

Il codice Morse; Cos'è

Il codice Morse è un **sistema di codifica** che permette di rappresentare lettere, numeri e segni di punteggiatura attraverso una sequenza di **punti (·)** e **linee** (-), chiamati anche "**dit**" e "**dah**"(Come il codice Morse riduce il linguaggio scritto a punti e linee, la sua versione "vocale" riduce il parlato a due suoni vocalici.).
È stato uno dei primi metodi utilizzati per la comunicazione a distanza, soprattutto tramite telegrafo, e successivamente anche via radio.

Il codice Morse prende il nome da **Samuel Morse**, uno degli inventori del telegrafo elettrico, sviluppato negli anni 1830 e 1840. La versione originale (codice Morse americano) fu poi semplificata e internazionalizzata nella seconda metà del XIX secolo, dando origine al **codice Morse internazionale**, ancora in uso oggi.

02

Come funziona

Inizialmente, la definizione del codice Morse (e per definizione si intende la corrispondenza fra le varie sequenze di punti e linee e le lettere dell'alfabeto) sembra casuale come la disposizione dei tasti su una tastiera di computer.
Se si osserva meglio, però, non è proprio così. I codici più **semplici e più brevi** sono assegnati alle lettere di uso più frequente, come la **E** (·) e la **T** (-). Le lettere meno comuni, come la **Q** (--·-) e la **Z** (--··) hanno codici **più lunghi.**
Quasi tutti conoscono un po' di codice Morse; Tre punti, tre linee e tre punti rappresentano SOS, il segnale internazionale per chiedere soccorso. SOS non è un'abbreviazione di qualche frase, è semplicemente una sequenza di codice Morse facile da ricordare.

Uno svantaggio del codice Morse è che **non distingue fra maiuscole e minuscole**, ma, oltre a rappresentare le lettere, prevede codici anche per i numeri, mediante una serie di cinque punti o linee.

La parola chiave qui è **due**.
Due tipi di lampeggiamenti, due suoni vocalici, in realtà, due cose differenti qualsiasi, in opportune combinazioni possono trasmettere qualsiasi tipo di informazione.

A	•–
B	–•••
C	•–•
D	–••
E	•
F	••–•
G	––•
H	••••
I	••
J	••–––
K	–•–
L	•–••
M	––

N	–•
O	–––
P	•––•
Q	––•–
R	•–•
S	•••
T	–
U	••–
V	•••–
W	•––
X	–••–
Y	–•––
Z	––••

1	•––––
2	••–––
3	•••––
4	••••–
5	•••••
6	–••••
7	––•••
8	–––••
9	––––•
0	––––––

.	•–•–•–
,	––••––
?	••––••
:	–––•••
;	–•––•–
-	–••••–
/	–••–•
“	•–••–•
‘	•––––•
(	–•––•
)	–•––•–
=	–•••–
+	•–•–•
$	•••–••–
¶	•–•–••
_	••––•–

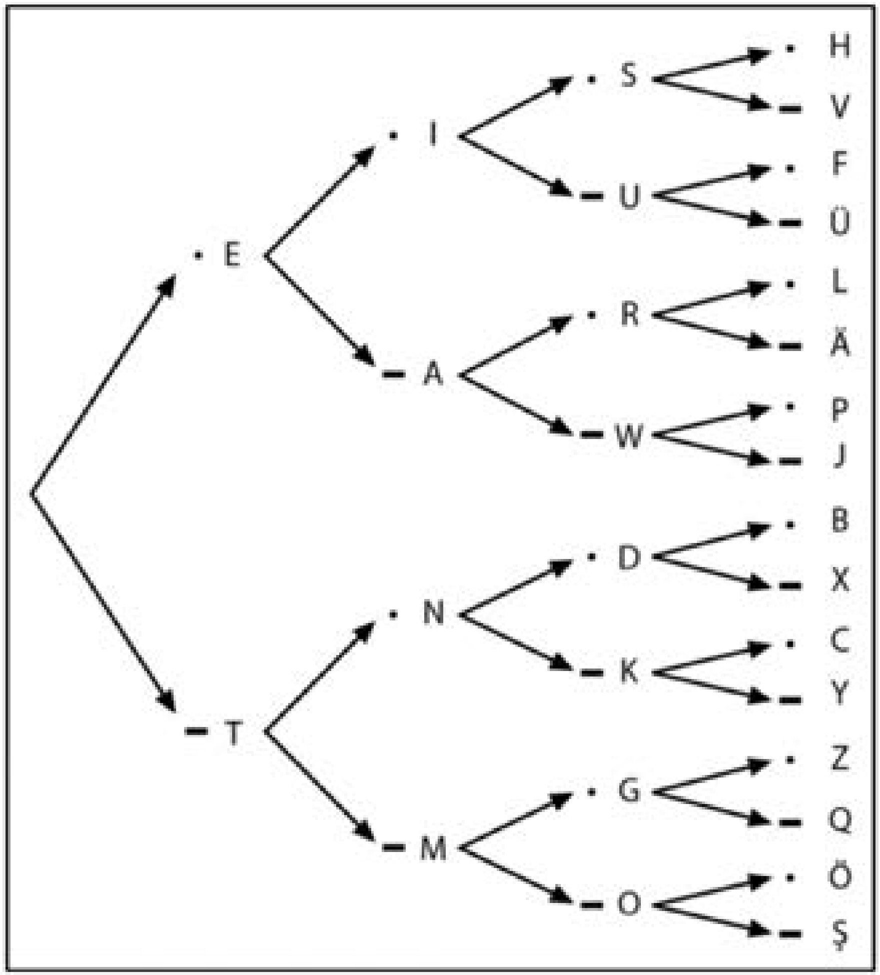
$$\text{numero di codici} = 2^{\text{numero di punti e linee}}$$

03

POV matematico

Dal **punto di vista matematico** nel codice Morse, il numero massimo di combinazioni possibili con n simboli segue sempre la formula
Essendo il codice Morse un codice **Binario** (quindi composto da due elementi) ogni combinazione è semplicemente **2 elevato alla potenza di punti e linee**

Noi potremmo continuare così all'infinito, per fortuna non è necessario scrivere tutti i codici possibili per stabilire quanti siano: tutto quello che dobbiamo fare è continuare a moltiplicare 2 per se stesso.
È un po' **come una moneta**, che può avere solo testa da una parte e croce sull'altra faccia. Se si lancia una moneta dieci volte si possono ottenere 1024 differenti sequenze di testa e croce. Le combinazioni di oggetti binari (come le monete) e di codici binari (come il codice Morse) si possono sempre descrivere con le potenze di due.



Lo stesso metodo è attuabile anche agli **interruttori** o al linguaggio di un calcolatore, detto **linguaggio macchina**, un codice binario, formato dai due soli simboli 1 e 0, segnali elettrici a 2 valori di tensione (VLOW, VHIGH), che rappresentano rispettivamente il circuito chiuso e quello aperto. Ai simboli 0 e 1 è stato dato il nome di **bit** (Binary digIT, cifra binaria). Parliamo di stati logici **0 (basso)** e **1 (alto)**.

Il bit è l'elemento minimo di memoria. Il bit si può trovare solo in due possibili stati che convenzionalmente sono rappresentati con le cifre 0 e 1.
Dominio di variazione del bit: {0,1}.
A livello elettronico questi due stati logici (0 e 1) corrispondono a "**circuito aperto**" (assenza di corrente elettrica) e a "**circuito chiuso**" (presenza di corrente elettrica).

Funziona come il circuito di una lampadina, chiusa o spenta

Codice ASCII

0	0011 0000
1	0011 0001
2	0011 0010
3	0011 0011
4	0011 0100
5	0011 0101
6	0011 0110
7	0011 0111
8	0011 1000
9	0011 1001
A	0100 0001
B	0100 0010
C	0100 0011
D	0100 0100
E	0100 0101
F	0100 0110
G	0100 0111
H	0100 1000
I	0100 1001
J	0100 1010
K	0100 1011
L	0100 1100
M	0100 1101
N	0100 1110

O	0100 1111
P	0101 0000
Q	0101 0001
R	0101 0010
S	0101 0011
T	0101 0100
U	0101 0101
V	0101 0110
W	0101 0111
X	0101 1000
Y	0101 1001
Z	0101 1010
a	0110 0001
b	0110 0010
c	0110 0011
d	0110 0100
e	0110 0101
f	0110 0110
g	0110 0111
h	0110 1000
i	0110 1001
j	0110 1010
k	0110 1011
l	0110 1100

m	0110 1101
n	0110 1110
o	0110 1111
p	0110 0000
q	0111 0001
r	0111 0010
s	0111 0011
t	0111 0100
u	0111 0101
v	0111 0110
w	0111 0111
x	0111 1000
y	0111 1001
z	0111 1010
.	0010 1110
,	0010 0111
:	0011 1010
;	0011 1011
?	0011 1111
!	0010 0001
'	0010 1100
"	0010 0010
(	0010 1000
)	0010 1001
space	0010 0000

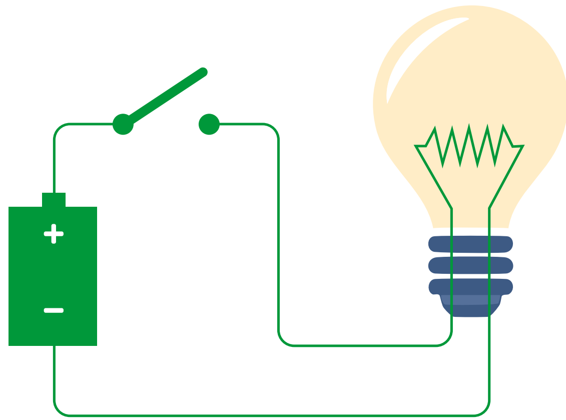
04

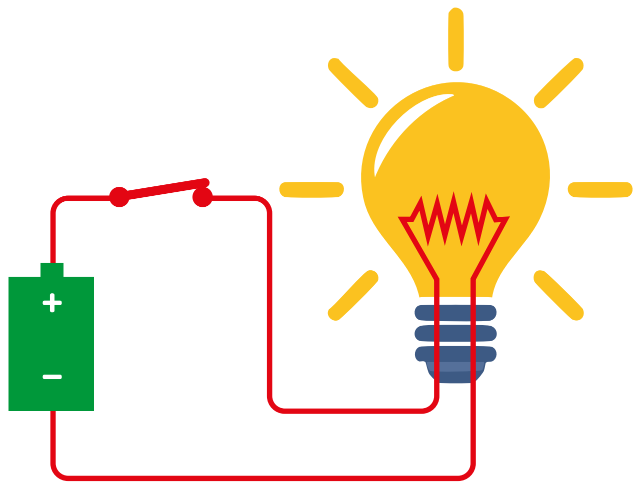
Come funziona un circuito

Il funzionamento dei circuiti elettrici di tutti i calcolatori moderni è basato su due stati elementari: la presenza oppure l'assenza di un **segnale elettrico**.
Per riuscire a visualizzarlo meglio, possiamo sftuttare la rappresentazione di un circuito elettrico di una **lampadina.**

All'interno dell'ampolla di vetro della lampadina si trova un filamento di tungsteno, che, quando gli viene applicata l'elettricità, si scalda e diventa luminoso. L'elettrictà in questo caso è alimentata da una batteria, collegata a dei sottili fili che trasportano l'elettricità, la lampadina riuscità ad accendersi solo nel momento in cui le due estremità dei fili solo collegate, l'oggetto che regola questi collegamenti è **l'interruttore**.

Se l'interruttore è aperto, la corrente non passa e non si accende la lampadina, se l'interruttore è chiuso, la lampadina si accenderà.





In questo modo si può creare un sistema Telegrafico bidirezionale

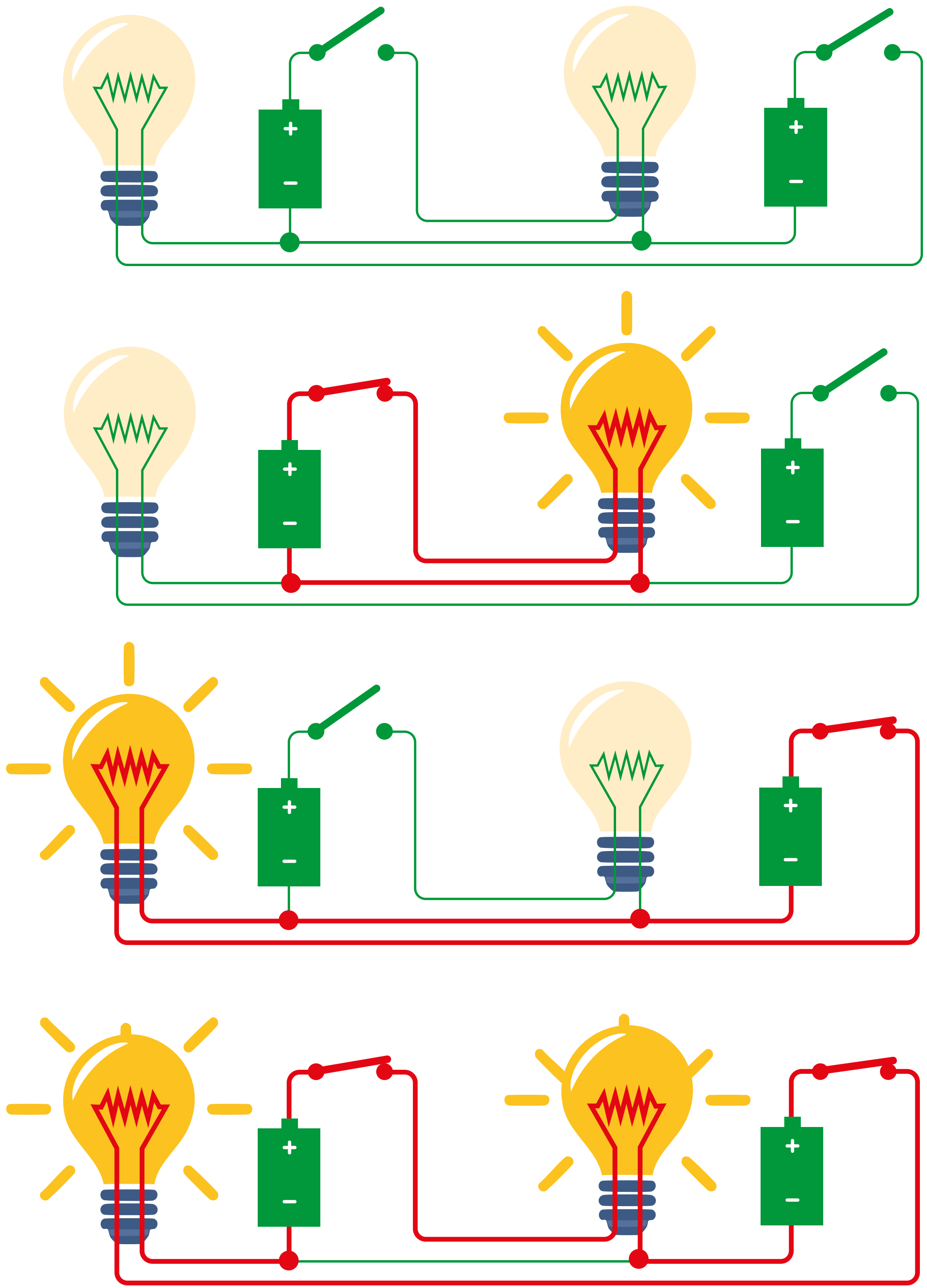
1\.
Tutti gli interruttori sono aperti ->

2\.
Se viene chiuso l'interruttore sinistro ->

3\.
Se viene chiuso l'interruttore destro ->

4\.
Se vengono chiusi entrambi gli interruttori ->

Cosa interessante, quando entrambe le lampadine sono accese, non passa corrente nel filo neutro del circuito. L'uso di un neutro per unire in un unico circuito due circuiti separati ci ha consentito di ridurre i collegamenti elettrici fra le due case da quattro fili a tre, e quindi di ridurre del 25 percento anche il costo dei nostri fili.

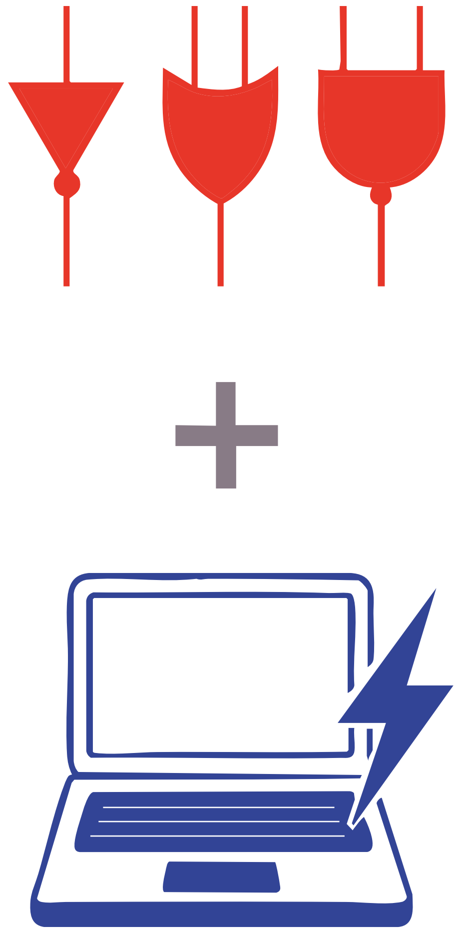


05

L’essenza pura dei computer

Se volessimo ridurre un computer alla sua essenza più pura, potremmo dire che **è il risultato dell’incontro tra due elementi fondamentali: l’algebra booleana e l’elettricità.** Da un lato, l’algebra booleana fornisce le regole logiche che permettono di rappresentare e manipolare l’informazione in forma binaria, attraverso operazioni come AND, OR e NOT. Dall’altro, l’elettricità è il mezzo fisico attraverso cui queste operazioni vengono effettivamente realizzate nei circuiti elettronici. Insieme, queste due componenti danno vita al cuore di ogni calcolatore: un sistema capace di elaborare dati, prendere decisioni logiche e compiere operazioni complesse partendo da combinazioni semplici di 0 e 1.

In fondo, tutta la potenza dell’informatica moderna si fonda su questa straordinaria **sintesi tra logica astratta e realtà fisica.**



06

George Boole e l'Algebra Booleana

George Boole nacque in Inghilterra nel 1815, in un contesto sociale poco favorevole. Figlio di un calzolaio e di una ex domestica, avrebbe dovuto seguire le orme dei genitori secondo le rigide gerarchie dell'epoca. Tuttavia, grazie alla sua curiosità e al sostegno del padre, appassionato di scienza e matematica, Boole riuscì a formarsi da autodidatta in latino, greco e matematica, raggiungendo livelli eccezionali. Nel 1849 fu nominato primo professore di Matematica al Queen's College di Cork, in Irlanda.

Mentre diversi studiosi, come Augustus De Morgan, cercavano di **tradurre la logica in termini matematici**, fu Boole a compiere il passo decisivo. Nei suoi due lavori principali — *Analisi matematica della logica* (1847) e *Indagine sulle leggi del pensiero* (1854) — pose le basi della logica matematica moderna. Boole credeva che il pensiero umano seguisse regole logiche e che queste potessero essere espresse in forma matematica. Anche senza condividere questa visione, il suo contributo resta fondamentale. Morì nel 1864, a soli 49 anni, per una polmonite contratta dopo essere corso sotto la pioggia per tenere una lezione.

La sua più grande eredità è l'algebra booleana, una nuova forma di algebra che oggi è alla **base dell'informatica.**

07

Algebra “normale”
VS
l’Algebra Booleana

Algebra “normale” :
(algebra elementare o tradizionale)

- Valori: numeri reali o complessi (es. 2, -3, 0.5, √2, π, ecc.)

- Operazioni principali: somma (+), sottrazione (–), moltiplicazione (×), divisione (÷), potenze, radici, ecc.

- Risultati: continui e numerici.

Esempio:

· 2+3=5

· $x^2+3x-4=0$ = equazione classica

Algebra booleana :

- Valori: solo due: 0 (falso) e 1 (vero)

- Operazioni principali:

AND (∧ / x): restituisce 1 solo se entrambi gli input sono 1

OR (∨ / +): restituisce 1 se almeno uno degli input è 1

NOT (¬): inverte il valore (¬1 = 0, ¬0 = 1)

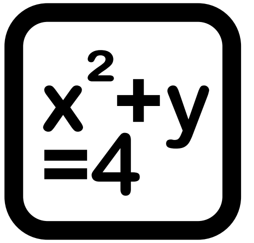
- Risultati: binari (0 o 1), usata in logica, informatica, elettronica digitale.

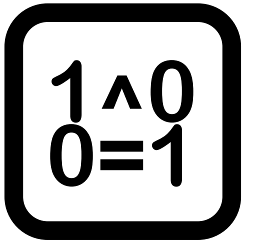
Esempio:

· $1 \land 0 = 0$

· 1 0 = 1

· $\neg 1 = 0$





08

L'Algebra Booleana

Il colpo di genio di Boole è stato rendere **l'algebra più astratta, separandola dal concetto di numero.** Nell'algebra booleana, gli operandi non stanno per numeri bensì per **classi.** Una classe è un gruppo di cose, simile a quello che in logica viene definito un **insieme.**

Nell'algebra convenzionale (numerica), si usano gli operatori + e x per indicare addizione e moltiplicazione. Nell'algebra booleana si usano i medesimi simboli, e questo po' creare confusione, tutti sanno come sommare e moltiplicare numeri nell'algebra tradizionale, ma come si sommano e si moltiplicano delle ***classi?***
In realtà nell'algebra booleana non si sommano né si moltiplicano classi; i simboli + e x significano qualcosa di completamente diverso. Nell'algebra booleana, il simbolo + significa **l'unione** di due classi, cioè la classe che ha come membri tutti i membri delle due classi di partenza.
Nell'algebra booleana il simbolo **x** indica invece **l'intersezione** di due classi, cioè la classe dove sono sono presenti solo tutti gli elementi che appartengono *sia* alla **prima** classe di partenza, *sia* alla **seconda.**
In più, come nell'algebra convenzionale, possiamo scrivere F x R come F R o semplicemente FR (come due aggettivi collegati).

Es.
M= gatti Maschi
F= gatti Femmine
R= di colore rosso
N= di colore nero

N + R = gatti o neri o rossi
F x R = gatti femmina di colore rosso /
FR = gatti "femmine rosse"

Per evitare confusione fra algebra convenzionale e algebra booleana, a volte si usano i simboli u e n per indicare rispettivamente l'unione e l'intersezione, al posto di + e x.

1

Per completare l'algebra booleana sono necessari tre altri simboli. Due di questi possono sembrare numeri, ma in realtà non lo sono, perché vengono trattati in un modo un po' diverso dai numeri.

Il simbolo 1,
nell'algebra booleana significa **"l'universo"**, cioè la classe di tutte le cose di cui parliamo.

Es.
1 = "la classe di tutti i gatti".

Quindi,
M + F = 1

Questo significa che se uniamo la classe dei gatti maschi con quella dei gatti femmina formiamo la classe di tutti i gatti.

-

Il simbolo 1 può essere usato **anche con un segno meno**, per indicare l'universo meno qualcosa.

Es.
1 - M

è la classe di tutti i gatti esclusa quella dei gatti maschi; la classe dei gatti femmina :
1 - M = F

0

L'ultimo che ci serve è **il simbolo 0 (zero)**, che nell'algebra booleana indica una **classe vuota**, a cui non appartiene alcun elemento. Questa classe ottiene quando si prende l'intersezione di due classi mutuamente esclusive, per esempio a classe di tutti i gatti che sono contemporaneamente maschi e femmine:
F x M = 0

Notate che a volte i simboli 1 e 0 funzionano nell'algebra booleana come nell'algebra convenzionale.

Es.
L'intersezione della classe di tutti i gatti e della classe delle gatte femmine è la classe delle gatte femmine:
1 x F = F

L'intersezione della classe vuota e della classe delle gatte femmine è la classe vuota:
0 x F = 0

L'unione della classe vuota e della classe delle gatte femmine è la classe delle gatte femmine:
0 + F = F

09

Esempio

Scenario:
Immaginiamo una scuola in cui gli studenti possono studiare diverse lingue.

Definiamo le seguenti classi:

F = studenti che studiano francese
I = studenti che studiano italiano
S = studenti che studiano spagnolo
Q = 1 = tutti gli studenti della scuola (l'universo)

Obiettivo:

Trovare gli studenti che ***non*** studiano né italiano né spagnolo, ma che studiano **francese.**

Passaggi booleani

· I + S = tutti gli studenti che studiano italiano o spagnolo

· Q - (I + S) = tutti gli studenti che NON studiano né italiano né spagnolo

· F x [Q - (I + S)] = gli studenti che studiano francese, ma non italiano né spagnolo

Formula finale
F x (1 - (I + S))

"Studenti che studiano francese ma non italiano né spagnolo."

Se vuoi sapere invece chi studia solo una lingua (es. solo francese, escludendo chi studia anche le altre), potresti usare:

F - (F x (I + S))

"Studenti che studiano francese, ma non anche italiano o spagnolo."

Bibliografia

num.	Titoli	Specifiche parti
01	Code; il linguaggio segreto di computer e software	dal Capitolo 1 al Capitolo 8 .
02	Il linguaggio binario – Ing. Daniele Corti	Capitolo 2 e 3 .
03	Il codice binario e l'algoritmo Università di Roma Tor Vergata	