

# Informatica: Elogio di Babele

Simone Martini

Dipartimento di Scienze dell'Informazione  
*Alma mater studiorum* • Università di Bologna  
[www.cs.unibo.it/~martini](http://www.cs.unibo.it/~martini)

*Didamatica 2007*

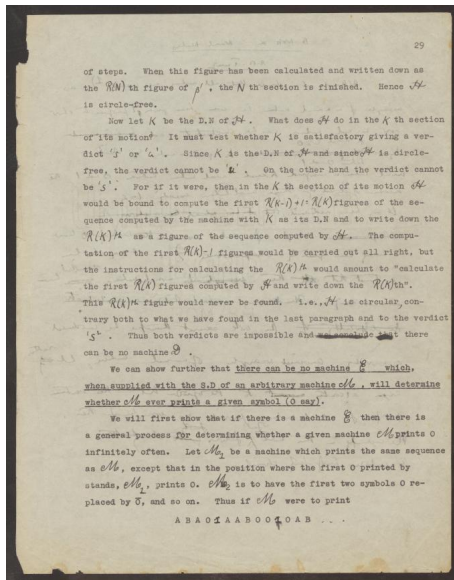


# Outline

- 1 Glosse ad alcuni padri fondatori



# Turing: macchine e linguaggio



*On Computable Numbers,  
with an application to the  
Entscheidungsproblem,  
Proc. Lond. Math. Soc. (2)  
42 pp. 230-265 (1936)*



# Linguaggio ed effettività

Una macchina:

- discreta
- basata su *un* linguaggio
- esprime il calcolabile
- incluso il proprio interprete

```
FO_072W=101001001001110100111010010100010111010000000101011101
00010001010110100010010110100000000010110100111010010010000101
11010000011010100000010101101001001110100001001001110100001000011
101011010000001101000000101011011010001001101000001010010110100
10001010000010010010110101000011010000101010111010100110100
0100000101110100000001101000100100011010000000111010000001101
110100010000110100001101000100110100100001011010010000000111010010
1010110101000010011101010000101100101000010110101000101011101
0010100001011010101001011101000000100110101000000011010100000001101010000
10101010000010001011010100101011101000001011010100001000011010
100100001101000101001110101000101010111010010000011101010010000101
11010010010011101010000010011101001010001101010100000101101010000
00110101000010010110100001001011101000001011101001000110101001001
01101000010110101000010110101001001011101010000000101110100101
0100101111010000000010111010001010101011101000000101110100001000
001110100000100011101010100001110101000100111010010100000101000
100001101000100000111010010000100111010001000101101000010010111
01000000010011010000100010110100100001011010010000011101001000
10010111010100100010111010010010001110101001000011101000000101011
101001010001011010000000001101001010101110100000100011101010000
00010101010000100110100000100110101000100101101001001011010010
001001110100001001101000001001101010001001011010001001011010010
0010011101001110101000010101101000010101101000010111010000100111
010000101011010100001010110101000010101101000010111010000100111
010000101011010100001010100001010100001011010100001011101000010011
01000010101101010000101010000101010000101101001101010000101110100
01000010110101000010101000010101000010110100001011101000010100100
11010100000011010001000001011000010001101010100001010000000110100010010
11010000100101101001000001010100010000101010100001010000000101
1010010100110101000000101010001010001010000101000010111010010100110
1010010110101000010101110101010111010000010110101010111010101
000111010011101000100101101000000010101101000100101101001000010
01010110100000101011101000010100110100000000101110100001000101
1010100100111010010100001010101000011010101000010111010000100111
01000010101101010000010100001001110100110100100000011010100
01000011101010000010101000010101010111010000101110100001011010010010
110101000000110100010000010110000100011010000000011010000000110100010010
11010000100101101001000001101010100001000101010100001010000000101
1010010100110101000000101010001010001010000101000010111010010100110
1010010110101000010101110101010111010000010110101010111010101
000111010011101000100101101000000010101101000100101101001000010
0101101000010101110100100001101010001110100100010000101101000000101
1010000010000101110100000011010000010010101110100000011101001010
01010111010000101101000010000010101000010011010000101000010100011
01000010010111010100001101000010100001010000101010000101010000100101
0110101000011101000100000010110101000011010000101000010101010101
00111010001000100101110101010011101001000000010111010000000101000
010010101011010000000110100100000101011010000100110100000100010110
011100010000001011101010000110101000100010110101010011101000010101
01011101000000010110101000011101010001010111010101001110101010001110
1010000101110100100011010100001011010001101000110100010010101010101
00111010010010100101110100101101000010101101000011010010101010101
000011010100001010100001010100001010100001010100001010100001010101
10000011010100000101110100001010100000101010000010101000001010101
00000101100101011101010001000010111010010101010001000101010001000101
010000101000000001011101000010100000001010111010010101110100000001
0010110100101011101010001010111010000110100001010011101010010101
010111010001110101010000001011101001110100111010000000100101
```



## L'unità delle lingue

- Non vi è calcolo senza linguaggio
- Un solo linguaggio esaurisce l'esprimibile
- È linguaggio finito (addirittura: due soli simboli)
- Tutti gli approcci all'effettività sono ridotti ad unità
- "Turing's compelling analysis"
  - La tesi di Church



## Babele...

- Altri formalismi per la calcolabilità:
  - ▶ funzioni ricorsive, schemi ricorsivi,  $\lambda$ -calcolo, sistemi di Post, URM, ...
- linguaggi *assembly*
- Fortran
- Algol 58, 60, 68, W
- Lisp
- Cobol
- PL/I
- Simula 67
- Pascal
- BCPL, B, C
- Smalltalk
- ...



## Reductio ad unum: compilazione

Ogni linguaggio è *equivalente* ad un qualsiasi altro

### Teorema

*Fissati due linguaggi di programmazione  $\mathcal{L}$  e  $\mathcal{L}'$  esiste un programma  $C_{\mathcal{L} \rightarrow \mathcal{L}'}$  tale che per ogni programma  $P$  scritto in  $\mathcal{L}$ ,  $C_{\mathcal{L} \rightarrow \mathcal{L}'}(P)$  è un programma in  $\mathcal{L}'$  *equivalente* a  $P$ .*



## Reductio ad unum: compilazione

Ogni linguaggio è *equivalente* ad un qualsiasi altro

### Teorema

*Fissati due linguaggi di programmazione  $\mathcal{L}$  e  $\mathcal{L}'$  esiste un programma  $C_{\mathcal{L} \rightarrow \mathcal{L}'}$  tale che per ogni programma  $P$  scritto in  $\mathcal{L}$ ,  $C_{\mathcal{L} \rightarrow \mathcal{L}'}(P)$  è un programma in  $\mathcal{L}'$  **equivalente** a  $P$ .*





## Compilatori meta-circolari

Un compilatore per un linguaggio, scritto in quello stesso linguaggio:  $C_{\mathcal{L} \rightarrow \mathcal{L}'}$ , scritto in  $\mathcal{L}$



Corrado Böhm,  
*Calcolatrici digitali:*  
*Du déchiffre de formules logico-*  
*mathématiques par la machine même*  
*dans la conception du programme*  
Ann. di Mat. Pura ed Applicata, vol. 37(4)  
pp. 175-217, 1954.



# Traduzione

- La diversità dei linguaggi è solo un epifenomeno della sottostante fondamentale unità
- *Ogni* linguaggio è equivalente ad un altro
- Cosa ci dicono i teorici della traduzione (tra lingue naturali)?



# Traduzione

- La diversità dei linguaggi è solo un epifenomeno della sottostante fondamentale unità
- *Ogni* linguaggio è equivalente ad un altro
- Cosa ci dicono i teorici della traduzione (tra lingue naturali)?



# Intraducibilità

- Non esiste una traduzione *fedele*
- Ogni lingua rappresenta un universo concettuale irriducibile ad ogni altro
- Neppure a livello di parole semplici, fattuali. . .



## George Steiner

*Una lingua riempie una cella nell'alveare delle percezioni e delle interpretazioni possibili. Articola una gerarchia di valori, di significati e di supposizioni che non corrisponde esattamente a quella di nessun'altra lingua. [...]*

*Parlando, creiamo mondi.*

*[G. Steiner, Errata, cap. 7, 1997]*



## Elogio di Babele

*Babele è stata il contrario di una maledizione. Il dono delle lingue non è una metafora vuota, è proprio un dono e una benedizione immensa.*

*[G. Steiner, Errata, cap. 7, 1997]*

*L'eros delle persone multilingui, persino di un monoglotta dotato di mezzi verbali e di un buon orecchio, è diverso da quello delle persone linguisticamente negate o stonate. [...]*

*Come doveva esser monotono l'amore in Paradiso.*

*[G. Steiner, Errata, cap. 7, 1997]*



## Elogio di Babele

*Babele è stata il contrario di una maledizione. Il dono delle lingue non è una metafora vuota, è proprio un dono e una benedizione immensa.*

*[G. Steiner, Errata, cap. 7, 1997]*

*L'eros delle persone multilingui, persino di un monoglotta dotato di mezzi verbali e di un buon orecchio, è diverso da quello delle persone linguisticamente negate o stonate. [...]*

*Come doveva esser monotono l'amore in Paradiso.*

*[G. Steiner, Errata, cap. 7, 1997]*



## E in informatica...

*Parlando, creiamo mondi.*

*[G. Steiner, Errata, cap. 7, 1997]*

Sorprendente: Alan J. Perlis, 1922 – 1990:

*A good programming language is a conceptual universe for thinking about programming.*





## E in informatica...

*Parlando, creiamo mondi.*

*[G. Steiner, Errata, cap. 7, 1997]*

Sorprendente: Alan J. Perlis, 1922 – 1990:

*A good programming language is a conceptual universe for thinking about programming.*



## Alan J. Perlis

*A language that doesn't affect the way you think about programming, is not worth knowing.*

*There will always be things we wish to say in our programs that in all known languages can only be said poorly.*



## Alan J. Perlis

*A language that doesn't affect the way you think about programming, is not worth knowing.*

*There will always be things we wish to say in our programs that in all known languages can only be said poorly.*



## Kenneth Iverson

Premio Turing 1979:

*For his pioneering effort in programming languages and mathematical notation resulting in what the computing field now knows as APL, for his contributions to the implementation of interactive systems, to educational uses of APL, and to programming language theory and practice.*



© Vector, The Journal of the British APL Association



## APL, circa 1964

- La somma degli elementi di un vettore:  $(+/\omega)$
- La media degli elementi di un vettore:  $(+/\omega) \div \rho\omega$
- I numeri primi da 1 a  $R$ :  $(\sim R \in R \circ. \times R) / R \leftarrow 1 \downarrow \iota R$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ \times R) / R \leftarrow 1 \downarrow \iota R$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ . \times R) / R \leftarrow 1 \downarrow \iota R$$

$$[1, 2, 3, 4, \dots, R]$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ \times R) / R \leftarrow 1 \downarrow \iota R$$

$$[2, \dots, R]$$





## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ . \times R) / R \leftarrow 1 \downarrow \iota R$$

$$R := [2, \dots, R]$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ \times R) / R \leftarrow 1 \downarrow \iota R$$

$$R := [2, 3, 4, \dots, R]$$

$$R \circ \times R = \begin{array}{cccccc} & 1 & 2 & 3 & \dots & R \\ & 2 & 4 & 6 & \dots & 2R \\ R \circ \times R = & 3 & 6 & 9 & \dots & 3R \\ & \vdots & \vdots & \vdots & & \vdots \\ & R & 2R & 3R & \dots & R^2 \end{array}$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ . \times R) / R \leftarrow 1 \downarrow \iota R$$

$$R := [2, 3, 4, \dots, R]$$

$$R \circ . \times R = \begin{array}{cccccc} & 1 & 2 & 3 & \dots & R \\ & 2 & 4 & 6 & \dots & 2R \\ R \circ . \times R = & 3 & 6 & 9 & \dots & 3R \\ & \vdots & \vdots & \vdots & & \vdots \\ & R & 2R & 3R & \dots & R^2 \end{array}$$

$$R \in R \circ . \times R = [0, 0, 1, \dots, 0]$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ \cdot \times R) / R \leftarrow 1 \downarrow \iota R$$

$$R := [2, 3, 4, \dots, R]$$

$$R \circ \cdot \times R = \begin{array}{cccccc} & 1 & 2 & 3 & \dots & R \\ & 2 & 4 & 6 & \dots & 2R \\ R \circ \cdot \times R = & 3 & 6 & 9 & \dots & 3R \\ & \vdots & \vdots & \vdots & & \vdots \\ & R & 2R & 3R & \dots & R^2 \end{array}$$

$$\sim R \in R \circ \cdot \times R = [1, 1, 0, \dots, 1]$$



## Un esempio

I numeri primi da 1 a  $R$ :

$$(\sim R \in R \circ . \times R) / R \leftarrow 1 \downarrow \iota R$$

$$R := [2, 3, 4, \dots, R]$$

$$R \circ . \times R = \begin{array}{cccccc} 1 & 2 & 3 & \dots & R \\ 2 & 4 & 6 & \dots & 2R \\ 3 & 6 & 9 & \dots & 3R \\ \vdots & \vdots & \vdots & & \vdots \\ R & 2R & 3R & \dots & R^2 \end{array}$$

$$\sim R \in R \circ . \times R = [1, 1, 0, \dots, 1]$$

$$[2, 3, \dots]$$



# Notation as a tool for thought

Turing Lecture

Kenneth Iverson, *Notation as a tool for thought*,  
CACM Vol. 33(8) pp. 444 - 465 (1980)

In limine:

*La quantità di significato compressa in piccolo spazio dai segni algebrici è un'altra circostanza che facilita i ragionamenti che siamo soliti fare col loro aiuto.*

*[Charles Babbage]*



# Caratteristiche positive della notazione

secondo Iverson



# Caratteristiche positive della notazione

secondo Iverson

- Facilità di espressione delle costruzioni, così come si presentano nei problemi

*Una notazione deve esprimere convenientemente non solo le nozioni che derivano immediatamente da un problema, ma anche quelle che derivano da analisi successive, generalizzazioni, specializzazioni.*





# Caratteristiche positive della notazione

secondo Iverson

- Facilità di espressione delle costruzioni, così come si presentano nei problemi
- Evocatività (*Suggestivity*)

*Una notazione è evocativa se la forma delle espressioni che si presentano in un insieme di problemi suggerisce espressioni collegate che trovano applicazione in altri problemi.*



# Caratteristiche positive della notazione

secondo Iverson

- Facilità di espressione delle costruzioni, così come si presentano nei problemi
- Evocatività (*Suggestivity*)
- Capacità di subordinare i dettagli

*La brevità facilita il ragionamento. La brevità è ottenuta mediante la subordinazione dei dettagli.*



# Caratteristiche positive della notazione

secondo Iverson

- Facilità di espressione delle costruzioni, così come si presentano nei problemi
- Evocatività (*Suggestivity*)
- Capacità di subordinare i dettagli
- Economia

*L'utilità di un linguaggio aumenta col numero di argomenti che esso può trattare, ma diminuisce con le dimensioni del vocabolario e la complessità delle regole sintattiche.*



# Caratteristiche positive della notazione

secondo Iverson

- Facilità di espressione delle costruzioni, così come si presentano nei problemi
- Evocatività (*Suggestivity*)
- Capacità di subordinare i dettagli
- Economia
- Assoggettabile a dimostrazioni formali

*L'importanza delle dimostrazioni formali è chiara dal loro ruolo in matematica.*



# Notazioni

- Non esiste la notazione universale
- Domini diversi richiedono astrazioni diverse
- Padroneggiare più notazioni è possedere più “universi concettuali”



## Qualche considerazione “didattica”

Informatica:

- insieme di applicazioni
- tecnologia
- scienza

Una scienza eminentemente “basata su linguaggio”: Turing!



## Qualche considerazione “didattica”

Informatica:

- insieme di applicazioni
- tecnologia
- scienza

Una scienza eminentemente “basata su linguaggio”: Turing!



# Incontrare l'informatica

- Nel e attraverso il linguaggio
  - ▶ Astrazione
  - ▶ Effettività
  - ▶ “Programmazione”
  - ▶ Interazione
- Nella pluralità feconda dei linguaggi
- Che diviene polifonia di espressione





# Incontrare l'informatica

- Nel e attraverso il linguaggio
  - ▶ Astrazione
  - ▶ Effettività
  - ▶ “Programmazione”
  - ▶ Interazione
- Nella pluralità feconda dei linguaggi
- Che diviene polifonia di espressione



## Elogio di Babele

*Babele è stata il contrario di una maledizione. Il dono delle lingue non è una metafora vuota, è proprio un dono e una benedizione immensa.*

*[G. Steiner, Errata, cap. 7, 1997]*



## Sipario

*Come doveva esser monotono l'amore in Paradiso.*

*[G. Steiner, Errata, cap. 7, 1997]*

*Come doveva esser monotono **programmare** in Paradiso.*

