**ꓱSTORIA DELL’INFORMATICA**

**INTRODUZIONE**

La sua storia è caratterizzata da reazioni a catena di fissioni di problemi e fusioni di strumenti e soluzioni tra discipline diverse; una sorta di *percorso* ***evolutivo*** pilotato da ***problemi*** , ***linguaggi*** , ***tecnologia***. L’informatica emerge come una singolarità nella storia dell’evoluzione culturale, scientifica e tecnologica ed è stata alimentata nel suo formarsi da continui e fruttuosi scambi di evenienze particolari che si sono accumulate nel tempo. E’ la storia di una disciplina scientifica emersa da un intreccio fecondo di tentativi messi in atto per risolvere problemi (scientifici, economici, amministrativi) cui hanno partecipato numerose discipline: matematica e filosofia *in primis*, ma anche fisica, economia, ingegneria, astronomia, geografia e linguistica.

Studiare la storia dell’Informatica contribuisce alla conoscenza di questa disciplina mediante la conoscenza dei problemi e delle relative soluzioni che hanno portato alla sua emersione e al suo consolidarsi.

In un momento di così rapida evoluzione per la tecnologia dell’informazione è importante comprendere gli elementi costitutivi dell’informatica affinché, anche in questa disciplina, la conoscenza del passato possa aiutare a delineare lo sviluppo futuro.

Tutto l’argomento può essere svolto in quattro capitoli in ciascuno dei quali viene utilizzata una prospettiva specifica per illustrare uno degli aspetti fondamentali della disciplina: linguistico, strumentale, applicativo e lessicale.

La prospettiva linguistica. Storia del linguaggio visto come strumento di comunicazione tra due o più sistemi biologici, strumentali, naturali o manufatti. A partire dalla comparsa di regole e modalità di rappresentazione, comunicazione e memorizzazione dell’informazione (alfabeto, grammatica, sintassi, logica e i gerghi delle discipline scientifiche), questo percorso termina con la definizione formale di linguaggio di programmazione.

La prospettiva strumentale. Dalla storia degli ausili strumentali e concettuali utilizzati per facilitare e realizzare il trattamento dell’informazione (sistemi di numerazione, aritmetica, algebra, calcolatrici manuali e automatiche), questo percorso termina con la progettazione della macchina calcolatrice elettronica, digitale a programma memorizzato universalmente chiamata computer.

La prospettiva applicativa. Storia di esigenze, problemi e curiosità (scientifiche) che hanno provocato e trainato ricerche specifiche che con l’accumulo dei loro risultati hanno contribuito alla nascita dell’informatica. Questo percorso, iniziato in modo diffuso in ogni disciplina in cui è fondamentale saper trattare l’informazione, termina con la constatazione che l’informatica costituisce l’infrastruttura fondamentale per sostenere lo sviluppo scientifico, culturale, economico e tecnico del mondo globalizzato.

La prospettiva lessicale. Storia di quattro aggettivi i cui significati, maturati con l’evolversi della cultura, hanno acquisito valori specifici e non ambigui nel contesto della disciplina informatica: digitale, automatico, effettivo e intelligente.

**CAPITOLO PRIMO: I LINGUAGGI**

Da Platone a Turing

**PLATONE** (Filebo).

Tra tutte queste scienze (ἐπιστήμη) il primato spetta alla [dialettica](http://it.wikipedia.org/wiki/Dialettica), la più alta forma di conoscenza, «in grado di investigare la chiarezza, la precisione, e il massimo grado di verità». Ma, se non sai calcolare non riuscirai a discutere del bene e del male e la tua vita non sarà quella di un uomo, ma quella di un’ostrica o di una medusa. I numeri governano il mondo, ma calcolare è una attività lenta, difficile e noiosa.

**ARISTOTELE** (Il sogno)

*Supponi che uno strumento con un comando potesse svolgere la sua funzione da solo in modo indipendente; ovvero supponete che le spolette di un telaio potessero essere autonome nel tessere e il plettro potesse pizzicare da solo le corde della cetra; in questo caso non ci sarebbe bisogno di alcun lavoro manuale e non ci sarebbe più la necessità di avere schiavi!!!*

**LULLO** (calcolo combinatorio)

Mentre per Aristotele i principi non si basano su dimostrazioni ma derivano dalla [esperienza](http://it.wikipedia.org/wiki/Esperienza) e dall'[induzione](http://it.wikipedia.org/wiki/Induzione), Lullo crede di risolvere ogni problema con precisione matematica: parte dal presupposto che ogni proposizione sia riducibile a termini e i termini complessi siano riducibili a più termini semplici o principi. Supposto di aver completato il numero di tutti i termini semplici possibili, combinandoli in tutti i modi possibili si otterranno tutte le proposizioni vere possibili: nasce così l'arte combinatoria, anche come forma di [mnemotecnica](http://it.wikipedia.org/wiki/Mnemotecnica), in quanto facilita la memorizzazione delle nozioni di base.

**LEIBNIZ** (calcolo filosofico).

**Sognatore. Si propone di costruire una ideografia universale per il pensiero (estensione di quella in uso per chimica e calcolo differenziale e integrale), definire uno strumento concettuale (un linguaggio) adatto per manipolare questi simboli e produrre argomentazioni in modo automatico.**

L’obiettivo di Leibniz era ridurre l’argomentazione a un calcolo, cioè costruire un sistema (un linguaggio) capace di descrivere ogni deduzione in modo da poterla sviluppare meccanicamente. Leibniz, influenzato dalla efficacia dei simboli da lui introdotti per descrivere il calcolo differenziale e integrale e suggestionato dall’uso del simbolismo binario illustrato nel cinese “I Ching” per descrivere situazioni, eventi e concetti, si propone di costruire un linguaggio utilizzabile come un sistema deduttivo nel quale ogni proposizione/teorema poteva essere ricavata/dimostrato, passo dopo passo, a partire da un numero limitato di assiomi.

Per ottenere questo obiettivo Leibniz ipotizza sia possibile trovare un linguaggio per descrivere deduzioni/argomentazioni basato su:

* creare un compendio/enciclopedia dei concetti rilevanti per descrivere la conoscenza,
* individuare le proprietà significative per descrivere questi concetti con simboli,
* trovare le regole per esprimere le deduzioni mediante manipolazioni di questi simboli.

***QUO FACTO … CALCULEMUS!***

Descritta la deduzione con questo linguaggio, il risultato si ottiene eseguendo un calcolo; così come si ottiene il risultato di una espressione aritmetica.

**BOOLE**

Boole inventa il primo linguaggio artificiale applicato alla logica.

L’algebra applicata alla logica è l’algebra ordinaria limitata ai due valori 0 e 1.

0\*X=0 1\*X=X

0+X = X 1+X=1

X = X\*X

------------------------------------------

X+(1-X) = 1

X\*(1-X) = X - X\*X = X – X = 0 *la contraddizion che non consente*!

Il principio di non contraddizione di Aristotele (non possono essere contemporaneamente vere una proposizione e la sua contraria) ottenuto come risultato di un calcolo!

Questa algebra può essere applicata per operazioni tra insiemi, per valutare la verità di proposizioni e per dedurre proprietà logiche (sillogismo).

Tutti gli X sono Y X\*(1-Y) = 0 <=> X = X\*Y

Tutti gli Y sono Z Y\*(1-Z) = 0 <=> Y = Y\*Z

----------------------

Tutti gli X sono Z X = X\*Y = X\*(Y\*Z) = (X\*Y)\*Z = X\*Z ⬄ X = X\*Z

X = X\*Z ⬄ X – X\*Z = 0 ⬄ X\*(1-Z) = 0

Tutte le deduzioni logiche prodotte con sillogismi possono essere descritte col linguaggio di Boole e essere ottenute con calcoli della sua algebra: questo è il primo risultato parziale positivo nella prospettiva ipotizzata da Leibniz, positivo, ma parziale. Infatti, la proposizione <tutti gli studenti bocciati sono pigri o stupidi> non è del tipo tutti gli X sono Y e non può quindi essere scritta col linguaggio di Boole.

**FREGE**

Per ottenere un sistema logico per descrivere tutte le inferenze deduttive della matematica Frege introduce simboli speciali usabili per analizzare la struttura di una singola proposizione. Per questo introduce i quantificatori universale e esistenziale per cui le due proposizioni

Tutti i cavalli sono mammiferi e alcuni cavalli sono purosangue diventano

(x)(se x è un cavallo allora x è un mammifero)

ꓱ(x)(x è un cavallo e x è un purosangue).

La proposizione tutti gli studenti bocciati sono pigri o stupidi, posto

B(x) per x è uno studente bocciato

S(x) per x è uno studente stupido

P(x) per x è uno studente pigro

diventa

(x)(B(x)S(x) o P(x))

Da questi esempi si vede che Frege non sta solo elaborando un trattamento matematico della logica, ma di fatto sta creando un nuovo linguaggio artificiale (Begriffsschrift) col quale scrivere inferenze logiche come operazioni eseguibili su simboli in modo meccanico (cioè senza la necessità di capirne il significato).

Il linguaggio di Frege (pur contenendo un errore di principio superato da Russell) è un ulteriore passo avanti rispetto al progetto di Leibniz: consente, in linea di principio, di descrivere tutti i ragionamenti (matematici), ma non garantisce di raggiungere un risultato concreto in un tempo finito.

**RUSSELL**

Nei *Principia Mathematica* viene presentato un sistema di logica simbolica capace di realizzare il programma di Frege di riduzione dell’aritmetica alla logica pura senza incorrere nei paradossi.

Secondo Poincaré, nei *Principia Mathematica* si ammette la possibilità di ridurre la matematica a mero calcolo (il sogno di Leibniz).

*Principia Mathematica* usano un sistema formale, calcolo dei predicati, logica del primo ordine, e … Prolog.

**HILBERT (VON NEUMANN)**

Si può fare tutta la matematica con la logica del primo ordine (calcolo dei predicati, col Prolog, con linguaggi di programmazione)?

Per derimere la questione, Hilbert pone due problemi:

1. Dimostrare che la logica del primo ordine è completa: cioè dimostrare che ogni formula valida poteva essere derivata con la logica del primo ordine;
2. trovare un metodo che, data una formula della logica del primo ordine, sia in grado di decidere , in un numero finito di passi (***effettivi***) se la formula è o non è valida; in altri termini, dimostrare che il problema della decisione (***Enscheidungsproblem***) è risolubile con un algoritmo.

Hilbert immagina un linguaggio simbolico puramente formale (un linguaggio di programmazione?); quindi vuole dimostrare che per ogni inferenza valida esiste una derivazione della conclusione dalle premesse scritta, passo dopo passo, esprimibile in questo linguaggio.

(Frege/Russell/Hilbert, il calcolo dei predicati del primo ordine e, in prospettiva il Prolog)

Esempio. Siano date le due inferenze seguenti

1. Chiunque ami è felice, W ama S quindi W è felice
2. I predatori hanno denti aguzzi, i lupi sono predatori, quindi i lupi hanno denti aguzzi

Hilbert vuole dimostrare che se una inferenza è tale che per ogni interpretazione delle lettere presenti nelle formule tale che quando le premesse siano enunciati veri anche la conclusione è vera allora è possibile passare dalle premesse alla conclusione usando le regole del suo linguaggio (Frege/Russell/Hilbert…il Prolog).

In questo caso, il *programma* per eseguire la derivazione è il seguente.

Le inferenze a) e b), posto

1. A(U,V) [con A(U,V) interpretabile come U ama V oppure come U preda V]
2. F(U) [interpretabile come U è felice oppure come U è un predatore]

posso essere riscritte in questo modo:

1. Per ogni x se esiste un y tale che A(x,y ) allora F(x),

Il *calculemus* di Leibniz/Frege/Russell/Hilbert è il seguente

(Ax))((Ex) A(x,y) => F(x))

A(U,V)

-------------------------------

F(U)

*Wir muessen wissen, wir werden wissen.*

**GOEDEL**

1. Goedel , nella tesi di dottorato, risolve positivamente il primo problema di Hilbert.

2. Dimostra (per assurdo scrivendo un programma!) che l’***Enscheidungsproblem*** non può avere soluzione positiva.

**TURING**

Introduce un linguaggio elementare per elaborare l’informazione (la Macchina Universale di Turing, prototipo di linguaggio di programmazione) esempio concreto formalmente definito di linguaggio alla Frege/Russell/Hilbert; con questo linguaggio è possibile scrivere programmi, algoritmi, macchine di Turing. Con questi strumenti si definisce la classe dei problemi calcolabili (risolubili con un programma, algoritmo, macchina di Turing) e mostra che esistono problemi non Turing-calcolabili; con questo viene mostrato costruttivamente che l’***Enscheidungsproblem*** non è risolubile.

Alan Turing: *On Computable Number with an Application to the Enscheidungsproblem*.

Alonzo Church: *An unsolvable Problem of Elementary Number Theory.*

============================================= ==========================================

**CAPITOLO SECONDO: Strumenti e metodi**

Dai sistemi di scrittura al calcolatore elettronico digitale a programma memorizzato

**AUTOMI MECCANICI**

**Da Erone a Jacquard**

**LE MACCHINE ANALITICHE**

**Babbage, Menabrea e Ada**

**MACCHINE TABULATRICI**

**Hollerith e l’industria elettromeccanica**

**MACCHINE ELETTROMECCANICHE**

**Zuse, Aiken e Stibitz**

**CALCOLATORI ELETTRONICI**

**Atanassoff, Bush, Mauchly, Eckert, Goldstine**

**IL COMPUTER**

**Architettura di von Neumann EDVAC e EDSAC**

=============================================================

**CAPITOLO TERZO: Esigenze e problemi**

**Dall’inizio della scrittura al sostegno della globalizzazione**

L’amministrazione dello stato

L’astronomia

La democrazia

Le esplorazioni geografiche

La rivoluzione industriale

La gestione dello stato (censimenti)

I conflitti mondiali (calcoli balistici e fluidodinamici)

La globalizzazione (internet di persone, cose e istituzioni, cloud computing, realtà virtuale).

=============================================================

**CAPITOLO QUARTO: Il vocabolario**

**Le parole chiave dell’informatica**

**Digitale**

**Automatico**

**Effettivo**

**Intelligente**

=============================================================

**CAPITOLO QUINTO: conclusioni e prospettive**

**La singolarità dell’informatica**

**L’intelligenza artificiale**

**Il sostegno alla globalizzazione (evoluzione delle professioni)**

**L’informatica nei processi educativi**

==============================================================================

Esempi di domande

Dal Big Bang emergono nell’ordine i linguaggi della fisica, della chimica e della biologia.

Da cellule capaci di replicarsi emergono sistemi viventi che sanno gestire linguaggi artificiali.

La scrittura alfabetica e i sistemi di numerazione inaugurano la rappresentazione digitale dell’informazione.

De Prony mette a punto un linguaggio matematico elementare per la tabulazione di polinomi.

L’invenzione di macchine capaci di produrre lavoro ha inaugurato l’era della automazione.

Muller ipotizza la trasformazione del telaio meccanico in macchina calcolatrice digitale, automatica e a programma memorizzato.

Babbage progetta la macchina alle differenze per eseguire la tabulazione di polinomi con la “macchina alle differenze”, primo esempio di macchina calcolatrice digitale meccanica per calcolare tavole numeriche di funzioni elementari utili per calcoli geodetici..

Babbage progetta la “macchina analitica”, primo esempio di macchina calcolatrice digitale meccanica a programma memorizzato.

Con le macchine di Babbage si realizza la prima grande fusione delle tre linee (linguistica, strumentale e applicativa) e viene anticipata l’emergenza di due categorie linguistiche che saranno associate ai due neologismi tipici dell’informatica: hardware e software.

Babbage progetta la macchina analitica per eseguire in modo automatico una sequenza di operazioni aritmetiche.

Menabrea descrive le potenzialità della macchina analitica di Babbage con esempi di programmi descritti con un semplice linguaggio di programmazione con istruzioni composte da un codice operativo e dai tre indirizzi degli elementi coinvolti nella singola operazione. Menabrea non descrive la struttura della macchina di Babbage, ma illustra con esempi l’uso del linguaggio che si deve usare per descrivere l’elaborazione che la macchina dovrà eseguire.

Ada Lovelace riconosce esplicitamente il contributo innovativo contenuto nel lavoro originale di Menabrea (redatto in francese) e lo traduce in inglese, arricchendo il testo con altri esempi di programmi.

Boole formalizza un linguaggio (l’algebra di Boole) che sarà utilizzato per progettare e costruire il computer, lo strumento base per tutte le applicazioni dell’informatica.

Macchine come protesi per facilitare il lavoro d’ufficio: calcolatrici (Schickard, Pascal, Leibniz, Poleni, …) e macchine da scrivere (clavicembalo scrivano, …).

Strumenti elettromeccanici per l’automazione del lavoro d’ufficio: da Hollerith alla produzione industriale delle macchine.

Strumenti linguistici per l’automazione delle deduzioni e delle dimostrazioni matematiche: da Leibniz a Turing.