STORIA LINGUISTICA

«La rivoluzione globale prodotta dall’informatica non è che l’ultima di una lunga serie di rivoluzioni dell’informazione: l’hanno preceduta, tra le altre, la scrittura, il linguaggio, la riproduzione sessuata, l’origine della vita, l’emersione delle «*costanti*» della fisica nate dal Big Bang. Ogni rivoluzione ha posto le basi per quella successiva, e tutte sono avvenute grazie alla capacità intrinseca dell’universo di elaborare informazioni. La vita, il cervello, la civiltà il computer **non sono apparsi nel mondo per puro caso» (\*)**

[Seth Lloyd, Il programma dell’universo, Einaudi].

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

INTRODUZIONE

Un *percorso* ***evolutivo*** pilotato da ***problemi*** , ***linguaggi*** , ***tecnologia***

La sua storia è caratterizzata da reazioni a catena di fissioni di problemi e fusioni di strumenti e soluzioni tra discipline diverse.

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 di risolvere problemi (scientifici, economici, amministrativi) cui hanno partecipato numerose discipline:

matematica, filosofia, linguistica, fisica, economia, ingegneria, astronomia, geografia.

Studiare la storia dell’Informatica contribuisce alla conoscenza dell’informatica mediante la conoscenza dei problemi e delle relative soluzioni che hanno portato alla sua emersione come disciplina scientifica.

Fornire elementi affinché anche in informatica, la conoscenza del passato possa aiutare a delineare lo sviluppo futuro.

1. La lettura linguistica. Storia del linguaggio visto come strumento di comunicazione tra due o più sistemi biologici, strumentali o naturali. 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.
2. La lettura 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 del computer.
3. La lettura 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 termina con la constatazione che l’informatica costituisce l’infrastruttura fondamentale per sostenere lo sviluppo scientifico, culturale, economico e tecnico del mondo globalizzato.
4. La lettura lessicale. Storia di quattro aggettivi il cui significato, maturato con l’evolversi della storia scientifica e culturale, ha acquisito un valore specifico e non ambiguo nel contesto della disciplina informatica: digitale, automatico, effettivo e intelligente.

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

5 CAPITOLI

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

ARISTOTELE

(concetti e categorie e sillogismo), Lullo (calcolo combinatorio), Leibniz (calcolo filosofico).

LEIBNIZ

L’obiettivo di Leibniz era ridurre l’argomentazione a un calcolo, cioè costruire un sistema capace di eseguire ogni deduzioni in modo meccanico. 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 in modo meccanico in ogni campo del sapere scientifico e filosofico a partire da un numero limitato di assiomi eseguendo effettive operazioni inferenziali.

Per ottenere questo obiettivo Leibniz propone un calcolo filosofico basato su:

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

*QUO FACTO … CALCULEMUS!*

yyt

BOOLE

Seguendo il progetto del *calculemus*  di Leibniz, Boole propone un linguaggio artificiale, un’algebra con due operazioni.

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

0+X = X 1+X=1

X+(1-X) = 1

X\*(1-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!

In questa algebra il simbolo \* può essere interpretato come simbolo della moltiplicazione tra numeri, dell’intersezione tra classi o della congiunzione tra proposizioni; quindi questo linguaggio può essere usato per descrivere 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\*(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 descritta 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

A(x)(se x è un cavallo allora x è un mammifero) A(x)(c(x)=>m(x))

E(x)(x è un cavallo e x è un purosangue) E(x)(c(x) AND p(x))

La proposizione tutti gli studenti bocciati sono pigri o stupidi col linguaggio di Frege può essere scritta

(A(x)(B(x)=>S(x) OR 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 è un ulteriore passo avanti (rispetto all’algebra di Boole) per il 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.

**La Begriffsschrift di Frege è il primo esempio di linguaggio formale artificiale dotato di sintassi e per questo si può considerare l’antesignano dei linguaggi di programmazione.**

**Con questo linguaggio era possibile descrivere tutti i ragionamenti dei matematici, ma …**

**Partendo da opportune premesse per raggiungere un risultato era possibile applicare le regole del linguaggio per tentare di raggiungere la conclusione, ma …**

**Se il tentativo falliva non c’era modo di sapere la causa, cioè se era applicato male il metodo o se era incompleta la premessa.**

RUSSELL

*Principia Mathematica* , definiscono un sistema formale, calcolo dei predicati del primo ordine, logica del primo ordine, (e … Prolog) e dimostrano che la formalizzazione completa della matematica entro un sistema di logica simbolica era possibile.

HILBERT (VON NEUMANN).

Vuole dimostrare che in questo linguaggio non è possibile dimostrare due formule che si contraddicano. E propone due problemi:

1. Dimostrare che la logica del primo ordine era completa, cioè che ogni formula valida poteva essere derivata nel sistema;
2. Entscheidungsproblem, problema della decisione, trovare un metodo che determini in un numero finito di passi **effettivi** se una data formula è o non è valida.

Questi due problemi di Hilbert sono una riproposizione formale del sogno di Leibniz: dimostrati questi due problemi la matematica si riduce a un calcolo (**quo facto ,,, calculemus**).

GOEDEL

Dimostra in positivo il primo problema di Hilbert e

Dimostra che il problema della decisione posto da Hilbert non può avere soluzione positiva.

TURING

Scompone il processo di elaborazione dell’informazione riducendolo al trattamento di singoli simboli e

introduce un linguaggio (di programmazione di basso livello) col quale definisce la classe dei problemi calcolabili (risolubili con un algoritmo) e dimostra che il problema della decisione di Hilbert non è risolubile con un algoritmo.

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

Esempi che mostrano la storia dell’informatica come un processo evolutivo caratterizzato da fissioni di problemi e da fusioni fra soluzioni fra elementi provenienti da discipline diverse.

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 IBM.

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