Storia dell'Informatica - Seconda Parte

Appunti

Luca Barra

Anno accademico 2023/2024

1
Premessa

1.1 Licenza

Questi appunti sono rilasciati sotto licenza Creative Commons Attribuzione 4.0 Internazionale (per maggiori informazioni consultare il link: https://creativecommons.org/version4/). Sono basati sulle slides del corso "Storia dell'Informatica" del prof. Felice Cardone.

1.2 Formato utilizzato

In questi appunti vengono utilizzati molti *box*. Questa è una semplice rassegna che ne spiega l'utilizzo:

Box di "Concetto sbagliato":

Concetto sbagliato 1.1: Testo del concetto sbagliato

Testo contente il concetto giusto.

Box di "Corollario":

Corollario 1.2.1 Nome del corollario

Testo del corollario. Per corollario si intende una definizione minore, legata a un'altra definizione.

Box di "Definizione":

Definizione 1.2.1: Nome delle definizione

Testo della definizione.

Box di "Domanda":

Domanda 1.1

Testo della domanda. Le domande sono spesso utilizzate per far riflettere sulle definizioni o sui concetti.

Box di "Esempio":

Esempio 1.2.1 (Nome dell'esempio)

Testo dell'esempio. Gli esempi sono tratti dalle slides del corso.

Box di "Note":

Note:-

Testo della nota. Le note sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive.

Box di "Osservazioni":

Osservazioni 1.2.1

Testo delle osservazioni. Le osservazioni sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive. A differenza delle note le osservazioni sono più specifiche.

Box di "Teorema":

Teorema 1.2.1 Nome del teorema

Testo del teorema.

2 Introduzione

Il corso si divide in una serie di sezioni:

⇒ Macchine mai esistite (Knowledge Navigator, Dynabook, Memex): poichè la storia dell'informatica è una storia di idee, di pionieri e di visionari;

⇒ Il modo di organizzare i testi in rapporto all'inondazione informativa: workstation di Otlet (non realizzata), the mother of all demos (Doug Engelbart, 1968), gli ipertesti di Ted Nelson, libraries of the future;

⇒ La cybernetica: originata da Norbert Wiener, ci si concentra sugli aspetti matematici della neurofisiologia che portarono al modello formale di neurone (McCulloch e Pitts, 1943) utilizzato da Von Neumann per la definizione della sua architettura di calcolatore;

⇒ Gli hippies e la controcultura: si parla di come la controcultura abbia influenzato la nascita dell'informatica;

⇒ La semiotica: si parte in ordine cronologico dalla preistoria, passando per Lullo fino a Leibniz, per arrivare ai sistemi formali.

2.1 La preistoria dell'informatica

Domanda 2.1

Perchè studiare cosi indietro nel tempo?

Risposta: perchè si vuole caratterizzare il concetto di calcolo e si vogliono comprendere le abilità cognitive su cui tale concetto si basa.

Circa 110.000 anni fa si ha, in Cina, il primo esempio di astrazione con degli *ossi intagliati*. Essa è la produzione consapevole di una traccia relativamente stabile. Successivamente, intorno al 8000 a. C., sono stati usati dei *gettoni* in argilla. Presumibilmente questa è la nascita simultanea della scrittura e del calcolo. I gettoni vennero affiancati dalle *tavolette d'argilla*, modellate usando dei calchi (come un rudimentale libro contabile).

I gettoni:

⇒ Favoriscono la raccolta di dati;

⇒ Sono immediati da comprendere;

⇒ Permettono le operazioni aritmetiche come manipolazioni concrete.

Un ulteriore fenomeno è quello delle *bolle* che contengono dei gettoni. Venivano usate come registrazione di un contratto quando un proprietario di bestiame dava in affitto la sua mandria per la transumanza.

Oltre agli strumenti d'argilla vennero usati dei *bastoncini di legno* che venivano divisi in due metà: una ricevuta e un titolo di credito. Il titolo di credito si chiamava *stock* (da qui il termine "stock market") e la ricevuta si chiamava stub.

2.1.1 Semiotica

Nei meccanismi illustrati nella sezione precedente i segni hanno un ruolo importante.

Definizione 2.1.1: Semiotica

La *semiotica* è la scienza che studia i linguaggi e i segni che li costituiscono.

Corollario 2.1.1 Il linguaggio

In ogni linguaggio sono presenti tre componenti:

⇒ Chi produce i segni (studiati dalla pragmatica);

⇒ I segni prodotti (studiati dalla sintassi);

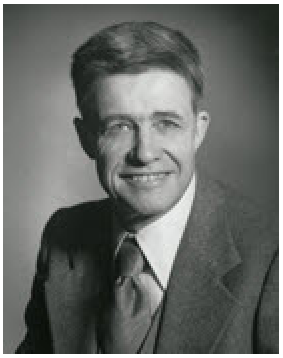
⇒ Il significato dei segni (studiati dalla semantica).

La semiotica è direttamente collegata all'informatica (computer semiotics):

⇒ Un computer è spesso utilizzato per generare, trasformare e visualizzare segni (il PC come medium);

⇒ Un computer viene programmato utilizzando un linguaggio.

Kenneth Iverson fu un importate semiotico e fu il fondatore del linguaggio esoterico APL, che nacque come linguaggio logico. APL influenzo molti linguaggi funzionali, come Haskell, e il paradigma di parallelismo.



2.2 Leibniz

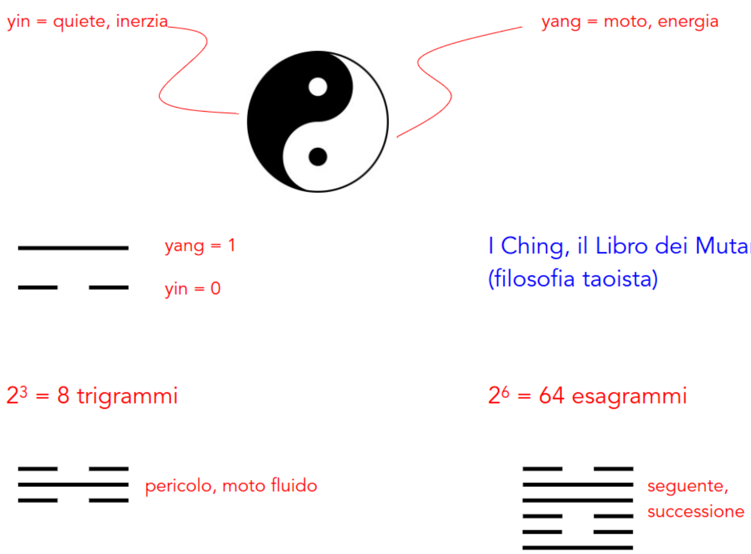
Nasce a Lipsia il 21 Giugno 1646, si laurea in Giurisprudenza nel 1666. I suoi primi scritti sono finalizzati al conseguimento di titoli accademici. Importante in questo periodo è la Dissertatio de Arte Combinatoria del 1666. Negli anni immediatamente successivi alla laurea diventa consigliere dell'Elettore di Magonza ed assume diversi incarichi politici.

Nel 1672 viene inviato a Parigi in missione diplomatica, per distogliere Luigi XIV dalla progettata invasione dell'Olanda ed invogliarlo invece alla conquista dell'Egitto. Fallita la missione, ottiene il permesso di fermarsi a Parigi (e Londra), dove rimane per 4 anni (marzo 1672-ottobre 1676) avendo la possibilità di conoscere la matematica e la fisica più avanzate. Nel 1676 scopre il calcolo infinitesimale (che verrà pubblicato solo ne 1684), già introdotto da Newton indipendentemente 10 anni prima. Nel 1705 inizierà tra i due una polemica che finirà solo con la morte di Leibniz. Nello stesso anno torna ad Hannover come bibliotecario presso il Duca di Hannover.

Tra il 1685 e il 1694: migliora la scatola di Pascal (*pascalina*) per l'addizione e la sottrazione per realizzare anche la moltiplicazione e la divisione (e l'estrazione di radice). La macchina, che opera mediante pulegge e ruote dentate, è conservata nella biblioteca di Hannover.

2.2.1 L'origine del bit

Leibniz fece un parallelismo tra i simboli dello yin e dello yang e i numeri binari. Le linee piene potevano essere associate al 1, mentre le linee vuote al 0. Così facendo si ottiene un sistema di numerazione binario.



Esempio 2.2.1 (L'interpretazione di Leibniz)

2.2.2 La caratteristica universale

La *caratteristica universale*, concepita come lingua o scrittura universale, si fonda sui seguenti principi:

⇒ Le idee sono analizzabili fino a idee semplici (atomiche);

⇒ Le idee possono essere rappresentate da simboli;

⇒ Le relazioni tra idee possono essere rappresentate da simboli;

⇒ Le idee si combinano mediante regole.

Definizione 2.2.1: Caratteristica universale

La *caratteristica universale* è un sistema di segni che rappresentano nozioni e cose, ma non parole. Le connessioni tra i segni rappresentano le relazioni tra le nozioni e le cose. Il nome di una nozione serve a:

⇒ Relazionarla con altre nozioni;

⇒ Relazionarla con lo schema dell'universo;

⇒ Indicare le esperienze necessarie per la conoscenza.

Note:-

L'apprendimento della lingua universale coincide con l'*enciclopedia*.

Il calculus ratiocinator

Definizione 2.2.2: Calculus ratiocinator

Il *calculus ratiocinator* è un insieme di tecniche per manipolare i segni della caratteristica universale. Può essere considerato come un *sistema formale*. Il frammento XX offre un'idea di questo calcolo: in esso vengono trattati molti assiomi e teoremi come il principio degli indiscernibili, la simmetria, la transitività, etc.

2.2.3 La mereologia

Definizione 2.2.3: Mereologia

La *mereologia* è la "dottrina delle parti".

Definizione 2.2.4: Principio di identità degli indiscernibili

Soo identici i termini che possono essere sostituiti a vicenda senza alterare la verità degli enunciati in cui compaiono. Si scrive A = B quando A e B sono identici.

$\Rightarrow$ $B + N = L$ significa che B e N compongono L, cioè che B è parte di L;

$\Rightarrow$ $A \leqslant B$ significa che A è parte di B.

Teorema 2.2.1 Simmetria
Se $A = B$ allora $B = A$.

Teorema 2.2.2 Transitività
Se $A = B$ e $B = C$ allora $A = C$.

Teorema 2.2.3
Se $A + B = B$ allora $A \leqslant B$.

Teorema 2.2.4
Se $A = B$ allora $A + L = B + L$.

2.3 Peano

Peano nacque a Cuneo il 27 agosto 1858. Si laureò in matematica nel 1880 e divenne professore di analisi matematica a Torino nel 1890. Nel 1889 pubblicò i suoi famosi *Principi di aritmetica*, in cui formulò un sistema assiomatico per i numeri naturali.

Definizione 2.3.1: Principi di aritmetica

I *Principi di aritmetica* sono un insieme di assiomi che definiscono i numeri naturali.

2.3.1 Sistemi formali

Note:-

Sono trattati più in dettaglio nel corso "Metodi formali dell'Informatica" e, in misura minore, nel corso di "Linguaggi e paradigmi di programmazione".

Definizione 2.3.2: Sistema formale

In un *sistema formale* si definiscono:

- ⇒ *Oggetti*, che sono tutti i termini costruibili a partire da atomi mediante operazioni;
- ⇒ *Proposizioni*, della forma $P(a_1, ..., a_n)$ dove P è un *predicato* e a_k sono termini;
- ⇒ *Regole di inferenza*, che permettono di dedurre nuove proposizioni. La conclusione di una regola di inferenza senza premesse è un'*assioma*.

$$\frac{A_1 \qquad \ldots \qquad A_v}{A}$$

Domanda 2.2

Come si usano le regole di inferenza?

Risposta: Per costruire derivazioni:

$$\frac{\dfrac{A_1 \qquad A_2}{A} \qquad B}{C}$$

Esempio 2.3.1 (Numeri naturali)

- ⇒ *Oggetti*: un atomo 0, un'operazione S, con un solo argomento;
- ⇒ *Proposizioni*: un predicato Num con un solo argomento;
- ⇒ *Regole di inferenza*: $\frac{}{\mathrm{Num}\{0\}}$ e $\frac{\mathrm{Num}\{n\}}{\mathrm{Num}\{S(n)\}}$.

2.3.2 Confronto tra logica e informatica

Logica:

- ⇒ Si studiano modelli di processi deduttivi;
- ⇒ Si studiano modelli di ragionamento;
- ⇒ Operatori modali.

Informatica:

- ⇒ Si studiano modelli di processi trasformazionali;
- ⇒ Si studiano modelli di calcolo (sequenziali, paralleli, distribuiti);

2.4 Kleene e le funzioni ricorsive

Nel 1981 Stephen Cole Kleene, matematico e logico statunitense, pubblicò l'articolo "*Origins of recursive function theory*" in cui descrisse l'attività di ricerca svolta da alcuni matematici e logici negli anni '30. In particolare, Kleene descrisse una serie di lezioni di Gödel tenute nel 1934 a Princeton sullo sviluppo di definizioni ricorsive primitive di funzioni numeriche. Lo *schema di ricorsione primitiva* permette di definire una funzione di numeri naturali f($\mathbf{x}$, n) a partire da funzioni predefinite b($\mathbf{x}$) e h($\mathbf{x}$, n, m) mediante lo schema:

$$f(x, 0) = b(x)$$

$$f(x, n+1) = h(x, n, f(x, n))$$

Inoltre si ha la composizione:

$$f(x) = h(g_1(x), \cdots, g_k(x))$$

Gödel usava la nozione di *sistema matematico formale* basata sulla nozione informale di *regola costruttiva* la cui applicazione si basava su una *procedura finita* del tipo necessario per calcolare funzioni definite mediante ricorsioni generali.

Nasce il problema della caratterizzazione delle ricorsioni ammissibili. Kleene aggiunse un *operatore di ricerca* non limitato che dato un predicato P(x, y) restituisce il valore minimo che soddisfa il predicato.

Nel 1938, Kleene ammise la possibilità di definire funzioni parziali.

2.5 Curry e la logica combinatoria

Nel 1927, Haskell Brooks Curry, matematico e logico statunitense, riscopri la nozione di *combinatore* (introdotta da Moses Schönfinkel nel 1920).

Definizione 2.5.1: Combinatore

Un *combinatore* è una funzione senza variabili libere.

Note:-

I sistemi di combinatori attuali usano K e S come combinatori, perchè sono sufficienti a generare tutti gli altri combinatori.

Inoltre riprende anche la possibilità di trattare funzioni a più argomenti come funzioni a un solo argomento che verrà chiamata *curryficazione*.

$$f(x, y) = f'(x)(y)$$

Nello stesso tempo, Alonzo Church, sviluppa il *lambda calcolo*.

Definizione 2.5.2: Lambda calcolo

Il *lambda calcolo* è un sistema formale per la definizione di funzioni intese come regole di corrispondenza. $\lambda x.M$ descrive la regola che assegna a un argomento x il valore M.

3
Informazione