

Storia dell'Informatica - Seconda Parte

Appunti

Luca Barra

Anno accademico 2023/2024



CAPITOLO 1	PREMESSA	PAGINA 1
1.1	Licenza	1
1.2	Formato utilizzato	1
CAPITOLO 2	DALLA PREISTORIA A VON NEUMANN	PAGINA 4
2.1	La preistoria dell'informatica Semiotica — 5	4
2.2	Leibniz L'origine del bit — 6 • La caratteristica universale — 7 • La mereologia — 8	5
2.3	Peano Sistemi formali — 9 • Confronto tra logica e informatica — 10	9
2.4	Kleene e le funzioni ricorsive	10
2.5	Curry e la logica combinatoria Paradosso di Russell e combinatori — 11	11
2.6	L'analisi dei processi di calcolo di Turing Non tutte le funzioni sono calcolabili da una macchina di Turing — 12	12
2.7	Von Neumann: il Computer come Organismo Artificiale Automi cellulari — 15 • La Vita Artificiale secondo Conway (Game of Life) — 16	14
CAPITOLO 3	INFORMAZIONE E CIBERNETICA	PAGINA 18
3.1	Shannon Il modello di comunicazione — 19	18
3.2	Cibernetica e neurofisiologia	19
3.3	Von Neumann e la termodinamica del calcolo	20
3.4	La cibernetica Il feedback e il flusso di messaggi nei sistemi — 22 • Osservatori — 23 • Comunicazione sincrona e asincrona — 24 • Circuiti asincroni — 24	22
CAPITOLO 4	BUSH ED ENGELBART, MEMEX E DEMO	PAGINA 26
4.1	Introduzione	26
4.2	Il Memex Problemi di organizzazione dell'informazione — 27 • Indicizzazione — 28 • Browser vs Search — 28 • Associazione — 29 • Le funzionalità del Memex — 29	26
4.3	Il progetto di Doug Engelbart: Demo L'idea di "Augmentation" — 31 • Classi di Strumenti di Aumentazione — 32 • Il Memex come strumento per la strutturazione simbolica — 33 • Utilità del computer — 34	30
4.4	Panoramica della DEMO	34

CAPITOLO 5	TED NELSON E LICKLIDER	PAGINA 37
CAPITOLO 6	PAUL OTLET E ALAN KAY	PAGINA 38
CAPITOLO 7	LETTURE	PAGINA 40
7.1	Come potremmo pensare	40
7.2	Douglas Carl Engelbart: Developing the Underlying Concepts for Contemporary Computing	40
7.3	Man-computer symbiosis	40
7.4	Alan Kay's Universal Media Machine	40
CAPITOLO 8	DOMANDE	PAGINA 42
8.1	Lullo, Leibniz e la storia del calcolo	42
8.2	Turing e la fisica del calcolo	42
8.3	Funzioni calcolabili e combinatori	42
8.4	"As we may think"	42
8.5	Engelbart e Nelson	42
8.6	Otlet, Lickider e Kay	42

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 "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 definizioni

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 "Meta-nota":

Meta-nota 1.2.1

Testo della meta-nota. Le meta-note utilizzate com note più personali per fornire riflessioni e motivazioni al perchè si sta facendo ciò che si sta facendo. Vengono usate per simulare lo stile espositivo del prof. Cardone che spazia in poco tempo su molti argomenti. Possono essere scollegate direttamente dal contesto in cui si trovano e ai fini degli appunti possono essere tranquillamente ignorate.

Box di "Teorema":

Teorema 1.2.1 Nome del teorema

Testo del teorema.

2

Dalla preistoria a von Neumann

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 così 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 *osso 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 importante semiotico e fu il fondatore del linguaggio esoterico APL, che nacque come linguaggio logico. APL influenzò 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.

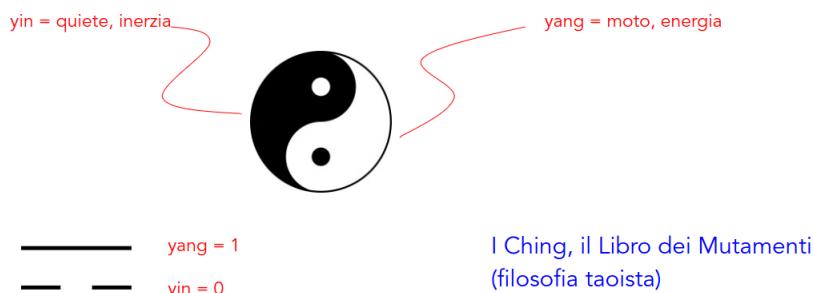


Nel 1672 viene inviato a Parigi in missione diplomatica, per distogliere Luigi XIV dalla progettata invasione dell'Olanda ed invogliarlo invece alla conquista dell'Egitto. Fallita la missione, ottiene il permesso di fermarsi a Parigi (e Londra), dove rimane per 4 anni (marzo 1672-ottobre 1676) avendo la possibilità di conoscere la matematica e la fisica più avanzate. Nel 1676 scopre il calcolo infinitesimale (che verrà pubblicato solo nel 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 puleggi 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.



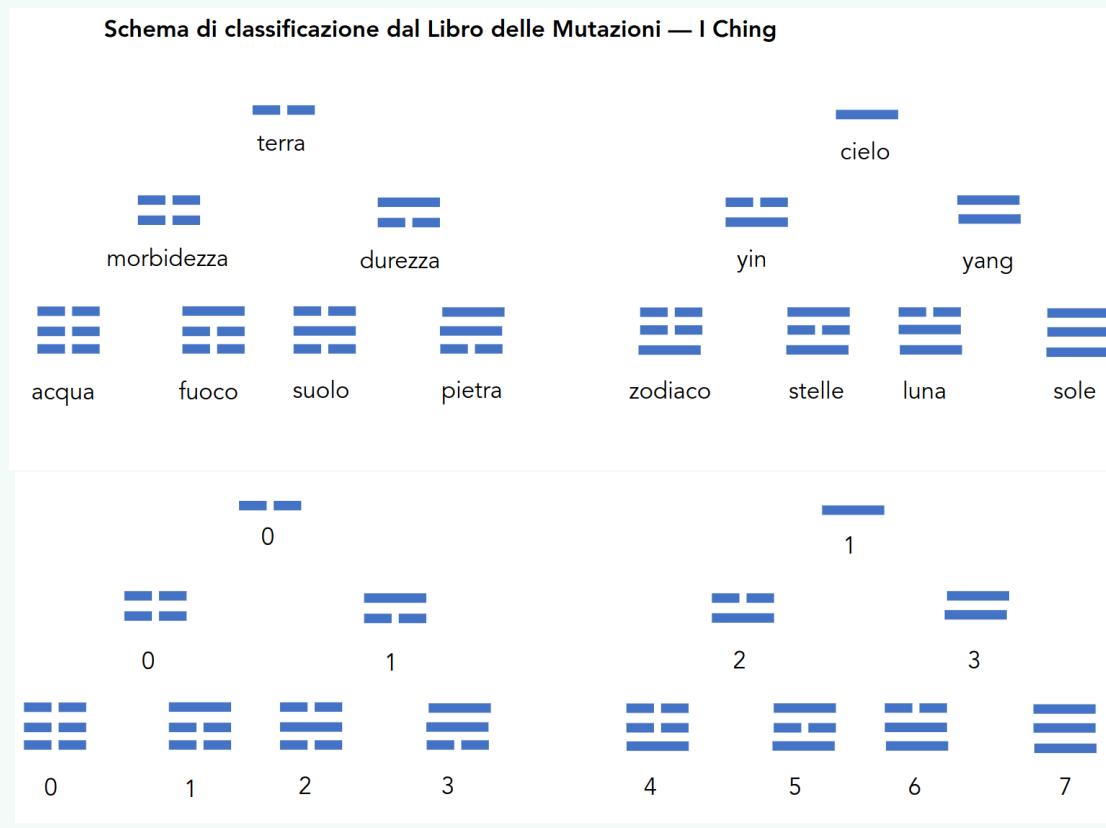
$2^3 = 8$ trigrammi

pericolo, moto fluido

$2^6 = 64$ esagrammi

seguente,
successione

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'*encyclopedia*.

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

Sono 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;

$\Rightarrow A \leq B$ 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 $A \leq B$.

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 $P(a_1, \dots, a_n)$ 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*.

$$\frac{A_1 \quad \dots \quad A_v}{A}$$

Domanda 2.2

Come si usano le regole di inferenza?

Risposta: Per costruire derivazioni:

$$\frac{\frac{A_1 \quad A_2}{A} \quad B}{C}$$

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*: $\frac{\text{Num}\{0\}}{\text{Num}\{S(n)\}}$ e $\frac{\text{Num}\{n\}}{\text{Num}\{S(S(n))\}}$.

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 1931 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 *sistema matematico formale* permette di definire una funzione di numeri naturali $f(x, n)$ a partire da funzioni definite $b(x)$ e $h(x, n, m)$ mediante lo schema:

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

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

Inoltre si ha la composizione:

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

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.

Esempio 2.5.1 (Combinatori)

$$\begin{aligned}\Rightarrow K(x, y) &= x; \\ \Rightarrow S(x, y, z) &= x(z)(y(z)); \\ \Rightarrow B(x, y, z) &= x(y(z)); \\ \Rightarrow C(x, y, z) &= x(z)(y); \\ \Rightarrow W(x, y) &= x(y)(y). \\ \Rightarrow I(x) &= x.\end{aligned}$$

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

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

Nello stesso tempo, Alonzo Church, sviluppa il *lambda calcolo*¹.

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 .

2.5.1 Paradosso di Russell e combinatori

$$WS(BWB) = Y = \lambda x.(\lambda z.x(zz))(\lambda z.x(zz))$$

Note:-

Si tratta del combinatore di punto fisso. Permette di definire funzioni ricorsive.

Definizione 2.5.3: Paradosso di Russell

$$(\lambda x.(\lambda z.x(zz))(\lambda z.x(zz)))\neg = (\lambda z.\neg(zz))(\lambda z.\neg(zz))$$

In cui $\{z|\neg(z \in z)\} \in \{z|\neg(z \in z)\}$. Ciò vuol dire che qualcosa appartiene a se stesso, ma non può appartenere a se stesso.

¹Visto approfonditamente nei corsi "Metodi formali dell'informatica" e "Linguaggi e paradigmi di programmazione", per cui in questi appunti non andrò in dettaglio dato che esula gli obiettivi del corso

2.6 L'analisi dei processi di calcolo di Turing

Nel paragrafo 9 del suo articolo del 1936, Turing introduce il comportamento di un *computer*. Lo esemplifica con un foglio di carta che viene astratto come un "nastro infinito" suddiviso in caselle (quadratini):

- ⇒ Il numero di simboli che si possono stampare è finito, perchè se ci fossero infiniti simboli ci si potrebbe confondere²;
- ⇒ Il comportamento è determinato dallo stato mentale e dal simbolo osservato, c'è un limite alla percezione delle caselle per lo stesso motivo di prima;
- ⇒ Le operazioni sono elementari, cioè non possono essere ulteriormente scomposte. Esse consistono in cambiamenti di stato mentale e del simbolo osservato³.

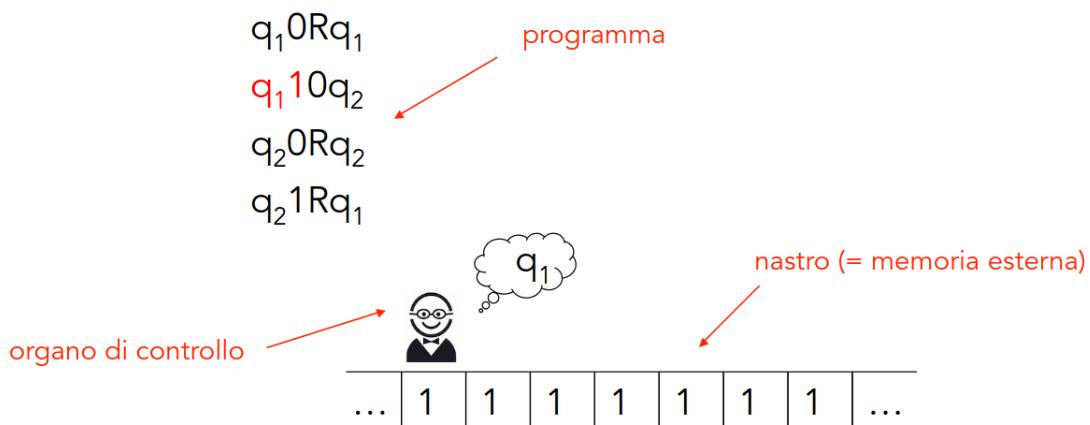


Figure 2.1: Macchine di Turing come Computer.

Definizione 2.6.1: Macchina universale

Una *macchina universale* (U) è una macchina di Turing che può simulare il comportamento di qualsiasi altra macchina di Turing (M) dato il suo programma (I_M) per ogni dato x :

$$U(I_M, x) = M(x)$$

È possibile che U sia più complessa di M . In informatica, il programma di U è chiamato *interprete*.

Note:-

Questo risultato suggerirà a von Neumann la possibilità di automi che generano automi di pari o maggiore complessità.

2.6.1 Non tutte le funzioni sono calcolabili da una macchina di Turing

Per dimostrare che non tutte le funzioni sono calcolabili da una macchina di Turing si può utilizzare la *diagonalizzazione di Cantor*:

- ⇒ Si immagina di poter numerare le funzioni da \mathbb{N} in \mathbb{N} ;
- ⇒ Si considera una funzione da \mathbb{N} in $\{0, 1\}^{\mathbb{N}}$;

²Un po' come il fatto che molti caratteri Unicode non possono essere usati nei nomi dei siti web in quanto troppo "simili".

³Un solo simbolo viene alterato.

- ⇒ Questo elenco può essere visto come una tabella;
- ⇒ Si vuole dimostrare che esiste una sequenza che non è in questa tabella;
- ⇒ Si prende la diagonale, che interseca tutte le righe in un punto;
- ⇒ Si cambiano tutti i valori della diagonale (0 diventa 1 e viceversa);
- ⇒ La diagonale trasformata può essere una riga dell'originale?
- ⇒ No, perché differisce da ogni riga in almeno un punto.

0	1	1	0	0	1	1	0	1	0	...
1	1	0	1	1	1	0	1	0	0	...
2	1	0	1	0	0	0	0	1	1	...
3	0	0	0	1	0	0	0	0	0	...
4	1	0	1	0	0	0	1	0	1	...
5	1	0	0	0	0	1	0	1	1	...
6	1	1	1	1	1	1	0	1	1	...
7	1	0	0	0	0	0	0	0	1	...
8	0	0	0	0	0	0	0	0	1	...
...

Figure 2.2: Diagonalizzazione di Cantor.

0	0	1	0	0	1	1	0	1	0	...
1	1	1	1	1	1	0	1	0	0	...
2	1	0	0	0	0	0	0	1	1	...
3	0	0	0	1	0	0	0	0	0	...
4	1	0	1	0	1	0	1	0	1	...
5	1	0	0	0	0	0	0	1	1	...
6	1	1	1	1	1	1	1	1	1	...
7	1	0	0	0	0	0	0	1	1	...
8	0	0	0	0	0	0	0	0	0	...
...

Figure 2.3: Diagonalizzazione di Cantor, inversione della diagonale.

Corollario 2.6.1 Problemi indecidibili

Non esiste una macchina di Turing M che, operando su un nastro che contiene:

⇒ La descrizione di una qualsiasi macchina di Turing T ;

⇒ Un input x ;

termina sempre i suoi calcoli scrivendo sul nastro il valore $M(T, x)$ dove:

⇒ $M(T, x) = 1$ se $T(x)$ termina;

⇒ $M(T, x) = 0$ se $T(x)$ non termina.

Note:-

Il problema della fermata è indecidibile.

2.7 Von Neumann: il Computer come Organismo Artificiale

Von Neumann, matematico e fisico ungherese⁴, fu influenzato dalle idee di McCulloch e Pitts⁵, che avevano definito un modello di neurone le cui reti potevano essere caratterizzate come *automi finiti*⁶.

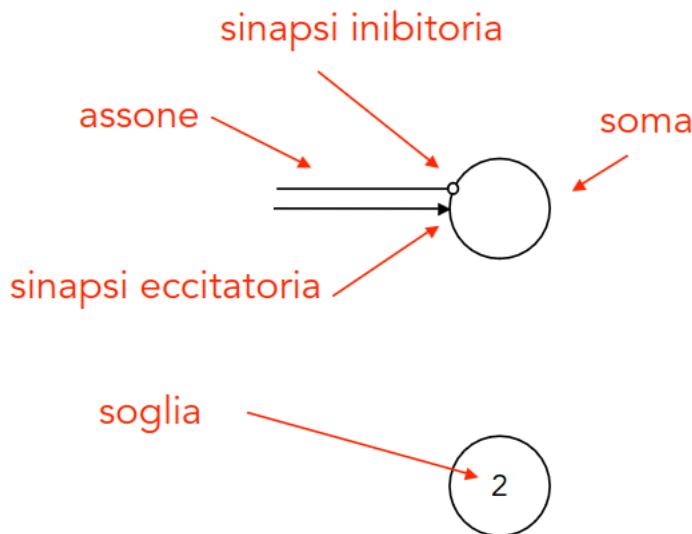


Figure 2.4: Il neurone di McCulloch e Pitts.

Nel 1945, von Neumann pubblicò il "First Draft of a Report on the EDVAC" in cui descriveva gli elementi di un calcolatore in termini "biologici" utilizzando il modello di McCulloch e Pitts:

⇒ *CPU*: neuroni di associazione;

⇒ *Input*: neuroni sensoriali;

⇒ *Output*: neuroni motori.

⁴Già visto nella prima parte del corso.

⁵A logical calculus of the ideas immanent in nervous activity.

⁶Notato da Kleene nel 1951

L'obiettivo di von Neumann era quello di *unificare*, tramite una teoria generale degli automi, il lavoro di Turing sulle macchine teoriche, il lavoro di McCulloch e Pitts sui neuroni e il lavoro di Shannon sulla teoria dell'informazione⁷.

Note:-

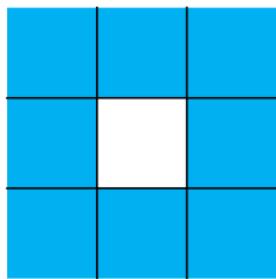
Questo tentativo verrà criticato dai neurofisiologi perché tentava di descrivere computer e cervello assiomatizzando il comportamento di *scatole nere*.

2.7.1 Automi cellulari

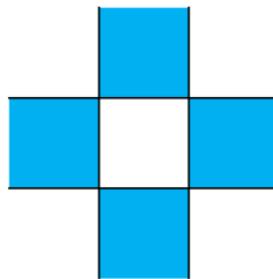
Negli anni '40, a Los Alamos, von Neumann e Ulam ebbero l'idea di automi che operino su una griglia di celle infinita (bidirezionale). Ogni cella ha un proprio stato e lo stato di una cella in un'istante T dipende dallo stato delle celle adiacenti in un istante $T - 1$. Il modello di von Neumann e Ulam aveva 29 stati.

Meta-nota 2.7.1

Da notare che la nozione di *celle adiacenti* è relativa: per von Neumann erano solo 4 (nord, sud, est, ovest), mentre per Moore erano 8 (aggiungeva anche le diagonali). Oltre a questo esistono altri modelli con celle esagonali e con forme diverse.



regola di Moore



regola di von Neumann

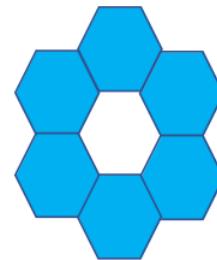


Figure 2.5: Ci sono differenti versioni.

Per von Neumann c'erano due modi di descrivere gli automi cellulari:

- ⇒ *McCulloch-Pitts (modo sintetico)*: strutture costruite a partire da elementi più semplici. Bisogna solo definire gli elementi e le loro connessioni (può essere complesso);
- ⇒ *Turing (modo integrale)*: si descrive, tramite assiomi, l'intero automa senza descrivere gli elementi da cui è composto, ma solo il loro comportamento.

⁷Si vedrà nel prossimo capitolo.

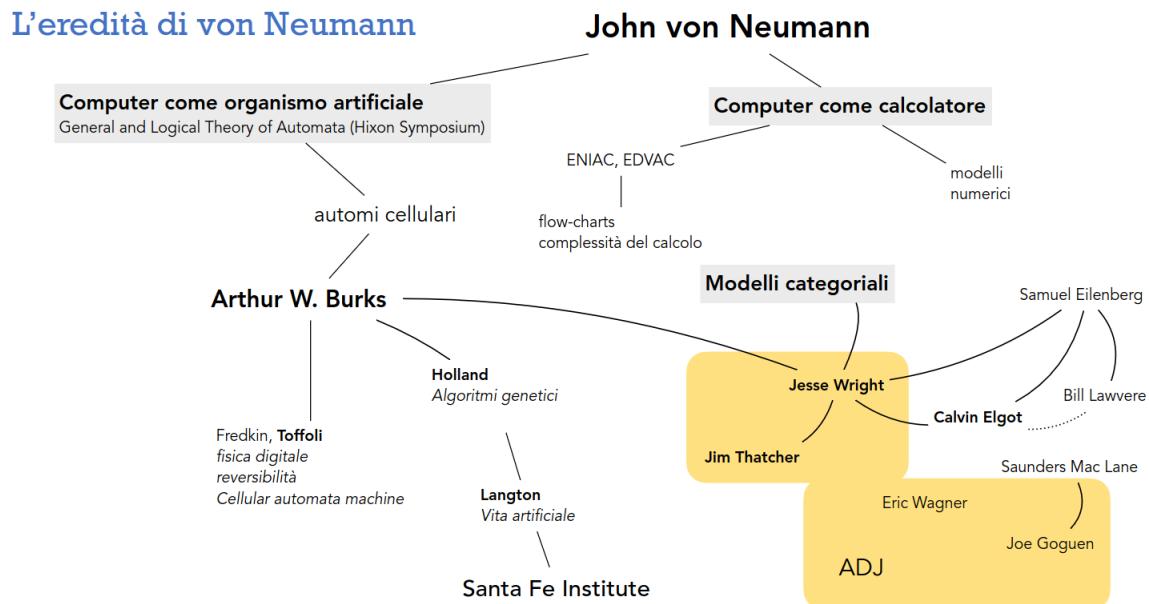


Figure 2.6: Sommario.

2.7.2 La Vita Artificiale secondo Conway (Game of Life)

Definizione 2.7.1: Game of Life

Il *Game of Life* è un automa cellulare ideato da John Conway nel 1970. Si basa su una griglia bidimensionale di celle quadrate, ognuna delle quali può essere in uno di due stati: vivo o morto. Le celle comunicano con le otto celle adiacenti. È un gioco a zero giocatori, cioè il suo sviluppo è determinato solo dallo stato iniziale.

Corollario 2.7.1 Regole

Le regole sono le seguenti:

- ⇒ Una cella morta con esattamente tre celle vive adiacenti diventa viva;
- ⇒ Una cella viva con due o tre celle vive adiacenti rimane viva;
- ⇒ In tutti gli altri casi la cella muore.

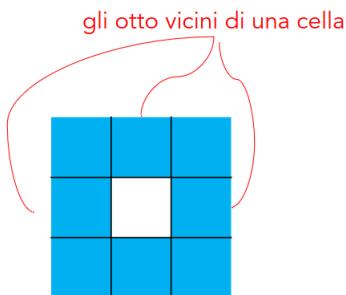


Figure 2.7: Game of Life.

Note:-

Una configurazione famosa è il *glider* (aliante), che si muove nella griglia.

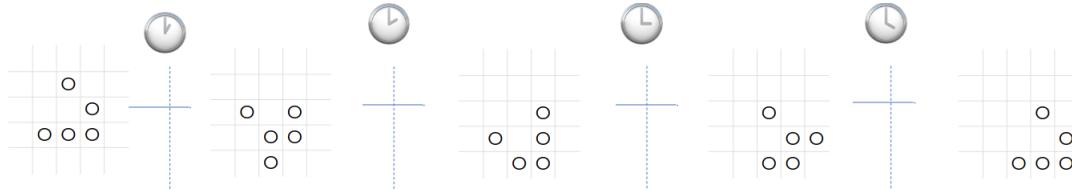


Figure 2.8: Glider.

3

Informazione e cibernetica

3.1 Shannon

Claude Shannon è un matematico che ha lavorato per la Bell Labs, e ha scritto un articolo nel 1948, *A Mathematical Theory of Communication*, in cui ha definito la teoria dell'informazione¹: "il problema fondamentale della comunicazione è quello di riprodurre ad un punto, esattamente o con qualche approssimazione, un messaggio scelto in un altro punto".

Shannon definisce un contesto tecnico di termini:

- ⇒ **Emittente ricevente**: il mittente è colui che invia il messaggio, il ricevente è colui che lo riceve.
- ⇒ **Messaggio**: è l'informazione che si vuole trasmettere.
- ⇒ **Rumore**: è tutto ciò che può interferire con la trasmissione del messaggio.
- ⇒ **Informazione**: è la quantità di incertezza che si riduce nel ricevente dopo aver ricevuto il messaggio.

Note:-

L'informazione in sé è considerata in base al numero di possibili messaggi che possono essere inviati.

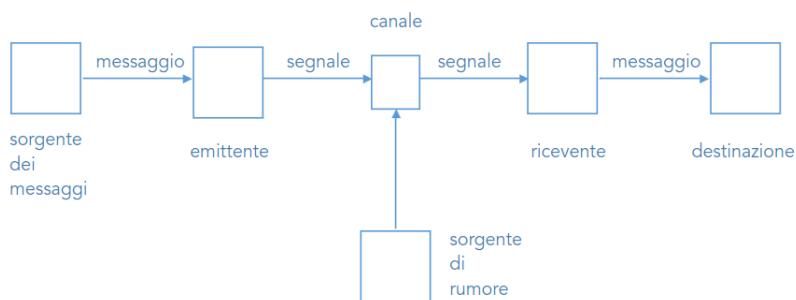


Figure 3.1: Schema di Shannon

¹Nella laurea magistrale è presente il corso "Teoria dell'Informazione" che approfondisce questi argomenti.

Meta-nota 3.1.1

L'informatica non è solamente uno scheletro che può essere riempito con qualsiasi cosa. Alla sua base vi sono idee e atteggiamenti fondanti che possono essere applicati in molti campi. Un esempio è la cibernetica che ha avuto una breve durata (circa 20 anni), ma ha avuto un impatto molto forte su molti campi. Esiste un'epistemologia dell'informatica, che ne studia le basi e le fondamenta.

Se si vuole approfondire l'argomento, nella prima parte del corso "Metodologie e Tecnologie Didattiche dell'Informatica" (MTDI o PREFIT) si ha una panoramica "motivazionale" sulle basi dell'informatica e sul suo essere una scienza.

3.1.1 Il modello di comunicazione

Il modello di comunicazione di Shannon purtroppo non mette in evidenza il fattore temporale e il ritardo. Quando si comunica in un contesto asincrono bisogna utilizzare un sistema di feedback per capire se il messaggio è stato ricevuto correttamente². Il modello di comunicazione di Shannon ha avuto varie interpretazioni, una delle quali è quella di Chapman.

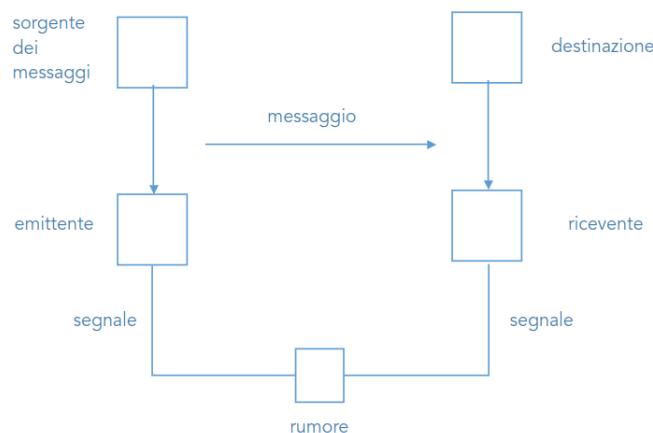


Figure 3.2: Modello di comunicazione di Shannon, reinterpretato da Chapman.

In questa interpretazione vengono identificati i livelli a cui i segnali possono esistere:

- ⇒ Nello strato basso è presente il rumore;
- ⇒ Al livello superiore è presente il messaggio.

3.2 Cibernetica e neurofisiologia

La cibernetica viene codificata da Norbert Wiener nel 1948, e si occupa di studiare i sistemi di controllo e di comunicazione nei sistemi biologici e nelle macchine.

Von Neumann aveva come obiettivo l'unificazione del lavoro di McCulloch e Pitts con il lavoro di Shannon e di Turing. Per Wiener, la nozione unificante era il feedback, mentre per von Neumann era la nozione di automa come elaboratore di informazione.

²Per esempio il sistema di ACK (Acknowledgement) di TCP visto nel corso "Reti I" / "Reti di calcolatori".

Lavori di von Neumann sugli automi:

- ⇒ *The general and logical theory of automata*, Hixon Symposium, 1948;
- ⇒ *Theory and organization of complicated automata*, serie di 5 lezioni all'università dello Illinois, 1949;
- ⇒ *Probabilistic logics and the synthesis of reliable organisms from unreliable components*, appunti delle lezioni alla Caltech, 1952;
- ⇒ *The Theory of Automata: Construction, Reproduction, Homogeneity*, 1952-1953;
- ⇒ *The computer and the brain*, Yale University Press, 1956 pubblicato postumo.

Per ricapitolare le relazioni tra i vari "attori":

- ⇒ *Shannon - Turing*: si incontrano nel 1934 ai Bell Labs, e discutono di comunicazione³;
- ⇒ *Wiener - Von Neumann - McCulloch - Pitts*: partecipano alle conferenze di Macy la cui nozione centrale è il messaggio (con le accezioni di Shannon);
- ⇒ *Wiener - McCulloch - Pitts*: membri del RLE;
- ⇒ *Von Neumann - Turing*: si incontrano nel 1937 a Princeton. Nel 1939 von Neumann offre a Turing un posto di lavoro come assistente a Princeton, ma Turing rifiuta;
- ⇒ *Shannon - Wiener*: sviluppano una teoria matematica della comunicazione basata sulla concezione dell'informazione/entropia di un insieme di messaggi.

Meta-nota 3.2.1

Pare che Shannon adorasse il monociclo e che spesso tenesse dei party in cui si esibiva in numeri di equilibrio.



Figure 3.3: Shannon sul monociclo

3.3 Von Neumann e la termodinamica del calcolo

Von Neumann cercò di valutare il costo energetico minimo per un atto elementare di generazione di informazioni. Da ciò deriva, nel 1949, la formula:

$$kT \log_e N$$

³Da ricordare Enigma e la crittografia.

- ⇒ k : costante di Boltzmann;
- ⇒ T : temperatura;
- ⇒ N : numero di alternative o possibili stati.

Questo calcolo attira le attenzioni di *Landauer*, un fisico teorico, che non riesce a dimostrarlo, ma osserva che le operazioni logicamente irreversibili (come l'operazione di cancellazione di un bit) generano entropia pari alla quantità di informazione cancellata. *Bennet*, un suo studente, nel 1973, dimostra che ogni calcolo può essere reso logicamente reversibile. *Fredkin* lavora a una base fisica per i calcoli reversibili.

Note:-

Da questo tipo di interessi nasce la computazione quantistica (che cosa succede se si considera la computazione come composta da operazioni meccaniche?).

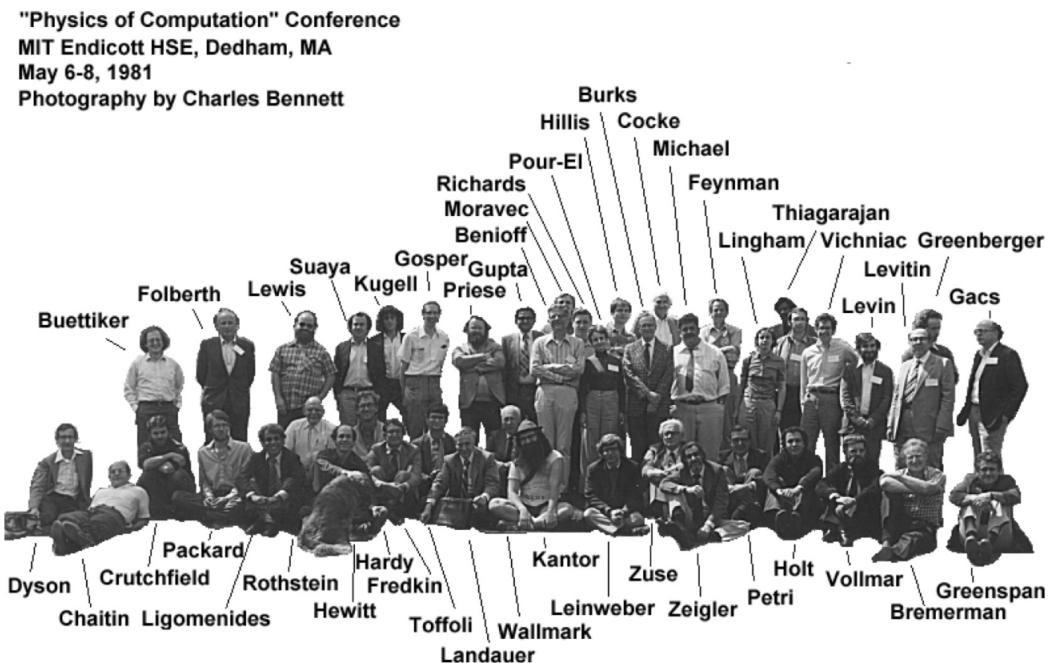


Figure 3.4: Physics of Computation

Meta-nota 3.3.1

- ⇒ *Bennet*: è un fisico teorico, non è direttamente presente nella foto perché fu lui a scattarla;
- ⇒ *Landauer*: è presente nella foto in primo piano;
- ⇒ *Fredkin*: è presente nella foto, vicino a Landauer e Toffoli;
- ⇒ *Toffoli*: fu uno studioso di fisica della computazione che interagi con Burks;
- ⇒ *Burks*: in secondo piano;
- ⇒ *Feynman*, *Dysan* e altri fisici famosi;
- ⇒ *Zuse*: già visto nella prima parte del corso, inventore dello Z1;
- ⇒ *Kantor*: inventore e specialista dell'informazione come principio ontologico;
- ⇒ *Holt*: esperto di cibernetica;
- ⇒ *Gosper*: inventore della configurazione ad aliante, uno dei primi "hacker".

3.4 La cibernetica

Quando si *comunica* con un'altra persona gli si trasmette un messaggio, e quando l'altra persona risponde, si riceve un messaggio che contiene informazioni accessibili a sé stessi e all'altro. Quando si *controllano* le azioni di un'altra persona gli si comunica un messaggio (in forma imperativa). Tutto ciò si riduce a uno *scambio di messaggi*.

Definizione 3.4.1: Cibernetica

La *cibernetica* può essere vista come una teoria della trasmissione dei messaggi e delle sue condizioni che presentano un successo.

Note:-

Il computer diventa un oggetto cibernetico, in quanto è inserito in un contesto di comunicazione con esseri umani o con altri computer.

3.4.1 Il feedback e il flusso di messaggi nei sistemi

Wiener confrontava l'attività umana con quella di figure che danzano sopra un carillon secondo un modello. Tuttavia il modello in esame è un modello predisposto in cui il loro passato non ha influenza sul loro futuro. Non c'è feedback.

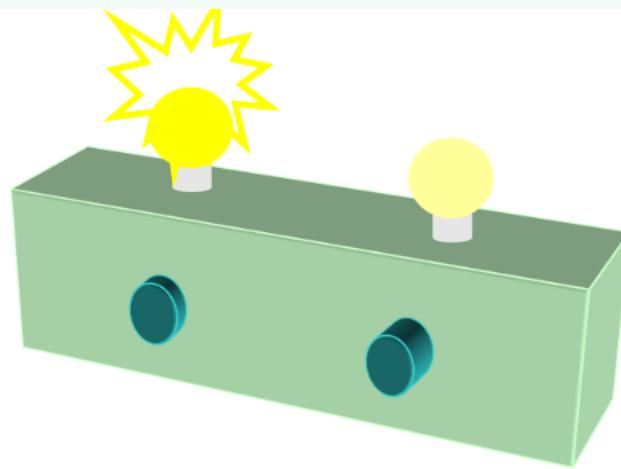


Figure 3.5: Modello IO

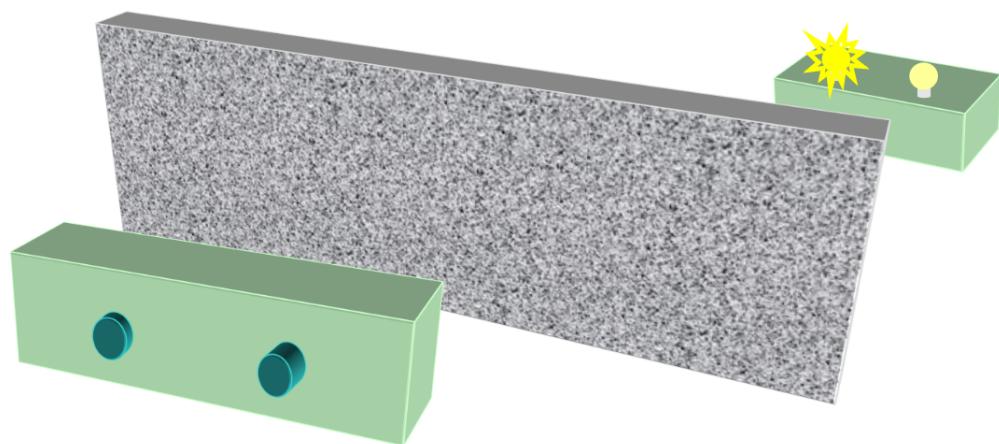
Holt propose l'idea che quando una persona scrive su una macchina premendo il tasto esso fornisce un feedback tornando indietro. La temporizzazione dei tasti è ciò che consente la scrittura. Questo si può estendere all'utente e all'interfaccia: la scelta di un'interfaccia comporta la progettazione dell'utente indicando cosa si può o non si può fare.

Esempio 3.4.1 (Scatola nera)

Si immagini una scatola con dei pulsanti che si possono premere. Quando si preme un pulsante si accende una luce. Prima di poter schiacciare un altro pulsante bisogna controllare che la luce si sia accesa. In questo caso la luce è il feedback.



Se i pulsanti sono premuti da un operatore che non può vedere le luci e le luci sono visibili a un altro operatore che non può premere i pulsanti, non si potrebbe stabilire una connessione tra luci e pulsanti e dunque la connessione non ha successo.



3.4.2 Osservatori

In alcuni sistemi, per esempio un termostato, è l'osservatore a concettualizzarlo. Nei sistemi di secondo ordine, l'osservatore è esso stesso parte del sistema osservato.

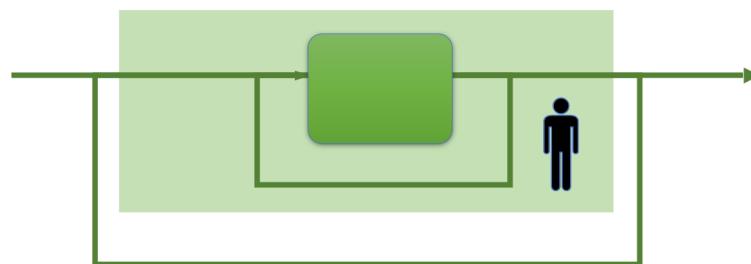


Figure 3.6: Osservatore

3.4.3 Comunicazione sincrona e asincrona

Definizione 3.4.2: Comunicazione sincrona

La *comunicazione sincrona* è una comunicazione in cui il mittente e il ricevente sono sincronizzati. In una comunicazione sincrona è presente un unico sistema di riferimento temporale condiviso da tutti i partecipanti.

Esempio 3.4.2 (Comunicazione sincrona)

- ⇒ Chiamata telefonica;
- ⇒ Conversazione faccia a faccia;
- ⇒ Sistema di posta elettronica in cui i messaggi impiegano n secondi per essere consegnati;
- ⇒ Segnali di fumo.

Definizione 3.4.3: Comunicazione asincrona

La *comunicazione asincrona* è una comunicazione in cui il mittente e il ricevente non sono sincronizzati. In una comunicazione asincrona non è presente un unico sistema di riferimento temporale condiviso da tutti i partecipanti.

Esempio 3.4.3 (Comunicazione asincrona)

- ⇒ Sistema di posta elettronica con garanzia di consegna entro n secondi;
- ⇒ Sistema postale ordinario;
- ⇒ Messaggi in bottiglia.

Note:-

Nella comunicazione asincrona è necessario un sistema di feedback per capire se il messaggio è stato ricevuto.

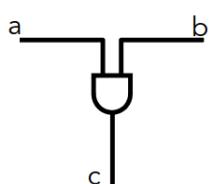
3.4.4 Circuiti asincroni

Definizione 3.4.4: Circuiti asincroni

I *circuiti asincroni* sono circuiti in cui i segnali di controllo non sono sincronizzati con un segnale di clock (orologio globale).

Corollario 3.4.1 C-Muller

Un elemento asincrono è il C-Muller per cui l'uscita è 1 se e solo se entrambi gli ingressi sono 1 e diventa 0 se entrambi gli ingressi sono 0.



Muller C-element:

if ($a = b$) **then** $c := a$

Figure 3.7: C-Muller

Note:-

Combinando più C-Muller si può costruire un circuito asincrono che implementa le micropipeline.

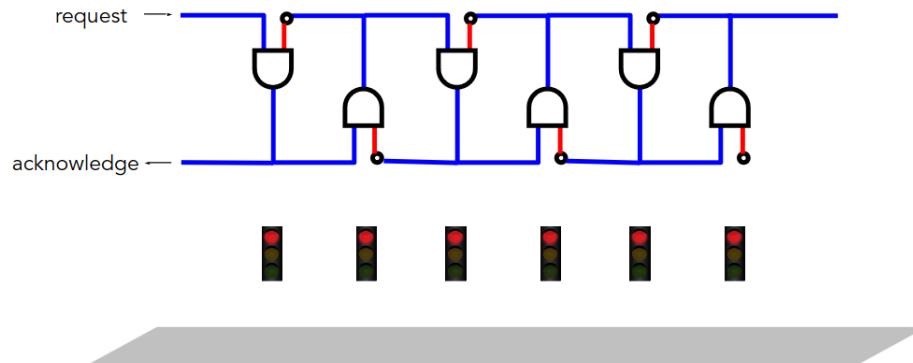


Figure 3.8: Micropipeline

Corollario 3.4.2 Reti di Petri

Grafi che generalizzano gli automati a stati finiti (DFA e NFA). Si hanno posti (condizioni) e transizioni (eventi).

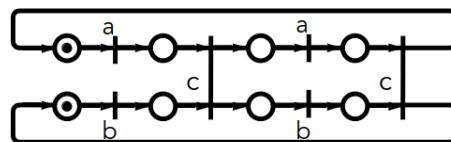


Figure 3.9: Rete di Petri

4

Bush ed Engelbart, Memex e Demo

4.1 Introduzione

Definizione 4.1.1: Knowledge Navigator

Un'idea di Apple, presentata nel 1987, di un assistente virtuale che aiuta l'utente a navigare tra le informazioni.

Il filmato di presentazione del Knowledge Navigator (1987):

- ⇒ *La comprensione del parlato*: il computer capisce il linguaggio naturale;
- ⇒ *La grafica e le finestre*: dietro questo video c'è Alan Kay, che ha lavorato a Xerox PARC e fu l'inventore delle finestre;
- ⇒ *Il touch screen*;
- ⇒ *La videochiamata*;
- ⇒ *La simulazione*: della desertificazione.

Domanda 4.1

Chi era Vannevar Bush?

Risposta: Vannevar Bush (1890 - 1974) era un ingegnere e scienziato americano. Lui teorizzò il Memex (non fu mai realizzato).

Note:-

Durante una conferenza in occasione del 50° anniversario di "As we may think" (1945) venne presentata un'animazione del Memex.

4.2 Il Memex

Definizione 4.2.1: Memex

Un sistema di archiviazione e ricerca delle informazioni, teorizzato da Vannevar Bush.

Note:-

Il Memex offre anche un'anticipazione di ciò che sarà l'*ipertesto*, che nascerà vent'anni dopo.

4.2.1 Problemi di organizzazione dell'informazione

Il problema che preoccupa Bush è *la perdita di informazioni* che si accumula continuamente nel tempo¹. Inoltre si aumenta progressivamente la specializzazione: le informazioni sono sempre più frammentate e sempre più specializzate (più difficili da comunicare). Bush ritiene che il problema non sia l'eccesso di pubblicazioni, ma il fatto che si siano estese oltre la capacità di gestione dei documenti. La *selezione*² è un problema per via delle enormi quantità di informazioni.

Esempio 4.2.1 (Addetto dell'ufficio informazioni)

L'addetto all'ufficio del personale di una fabbrica immette una pila di alcune migliaia di schede degli impiegati in una macchina selezionatrice, imposta un codice secondo una convenzione stabilita e produce in poco tempo una lista di tutti gli impiegati che vivono a Trenton e conoscono lo spagnolo.

Esempio 4.2.2 (Centralini telefonici automatici)

Si compone un numero e la macchina seleziona e connette solamente una tra un milione di possibili stazioni. Non le ispeziona tutte. Presta attenzione solo a una classe data dalla prima cifra, poi solo a una sottoclasse data dalla seconda cifra e così via; così procede rapidamente e quasi infallibilmente verso la stazione selezionata.

Problemi di indicizzazione:

- ⇒ Si cerca da sottoclasse a sottoclasse;
- ⇒ L'informazione si trova in un unico punto (a meno che non sia duplicata);
- ⇒ Bisogna avere regole per specificare il percorso, ma le regole sono complicate;
- ⇒ Quando si trova l'elemento bisogna riemergere dal sistema e rientrare attraverso un nuovo percorso.

Note:-

La situazione peggiore, ma più facilmente verificabile, è la ricerca in un albero.

Definizione 4.2.2: Classificazione

La *classificazione* è una segmentazione spaziale, temporale o spazio-temporale del mondo.

Un sistema di classificazione è un insieme di scatole in cui si mettono cose per fare un qualche tipo di lavoro.

Proprietà di un sistema di classificazione:

- ⇒ Ci sono principi univoci e consistenti;
- ⇒ Le categorie sono reciprocamente esclusive;
- ⇒ Il sistema è completo.

¹Questo problema non era nuovo. Era già stato affrontato da Paul Otlet.

²Processo di scelta delle informazioni.

Kingdom	Animalia
Phylum	Chordata
Subphylum	Vertebrata
Class	Mammalia
Subclass	Theria
Infraclass	Eutheria
Order	Carnivora
Suborder	Caniformia
Family	Canidae
Genus	<i>Canis</i>
Species	<i>Canis familiaris</i>
AKC Group	Hounds
AKC Breed	Beagles
Our Hero	Snoopy



Figure 4.1: Snoopy secondo due sistemi di classificazione.

4.2.2 Indicizzazione

Definizione 4.2.3: Indicizzazione

L'*indicizzazione* è un'operazione è l'azione di descrivere o identificare un documento (o un oggetto) nei termini del suo contenuto concettuale^a.

^aISO 5963.

Note:-

Lo scopo generale dell'indicizzazione è quello di rappresentare oggetti in modo che possano essere efficacemente trovati e utilizzati.

Corollario 4.2.1 Linguaggio di indicizzazione

Un linguaggio di indicizzazione è un sistema di rappresentazioni simboliche (un codice) che consentono la classificazione e la ricerca di documenti attraverso i codici assegnati ai concetti che essi contengono (indicizzazione per concetti).

Note:-

Ted Nelson, in "As we may think", criticherà il frantendimento per cui si vede nel lavoro di Bush un contributo alla information retrieval^a.

^aReperimento di oggetti informativi.

4.2.3 Browser vs Search

Le seguenti definizioni sono state date da Clay Shirky in "Ontology is Overrated. Categories, Links, and Tags" (2005).

Definizione 4.2.4: Browse

Browse significa che le persone fanno ontologia e categorizzazione, avendo la responsabilità di organizzare il mondo in anticipo.

Note:-

Il browse è uno schema passivo.

Definizione 4.2.5: Ricerca

Il paradigma della *ricerca* sostiene che nessuno può prevedere cosa sarà necessario in futuro. Quando se ne ha bisogno si tenterà di trovarlo basandosi sulla struttura di link disponibile.

Note:-

Per esempio, Google è un sistema di ricerca.

4.2.4 Associazione

La mente umana, secondo Bush, lavora per associazione: quando "afferra" un elemento scatta istantanemente a quello successivo in base a un'associazione di pensieri in accordo con una rete di percorsi determinata dai neuroni. La selezione per associazione può essere meccanizzata dal *Memex*.

Nel Memex vengono archiviati tutti i libri, le registrazioni e le comunicazioni. Esso è meccanizzato in modo da essere consultato con estrema velocità e flessibilità. Si tratta di un supplemento *personalizzato* ed allargato alla memoria dell'individuo.

Il Memex è uno strumento personale:

- ⇒ È utilizzato da una sola persona;
- ⇒ L'utilizzatore lo adatta ai propri interessi;
- ⇒ Il Memex non è collegato ad altri dispositivi (non esiste una rete di Memex).

Note:-

Il Memex è stato disegnato da Alfred D. Crimi, in Life.

Meta-nota 4.2.1

La versione del Memex su Life è stata letta da Engelbart in una baracca della croce rossa nelle Filippine. Engelbart rievocherà questa descrizione tra la fine degli anni '50 e l'inizio degli anni '60 quando deciderà di dedicarsi ad alcune problematiche di cui aveva avuto un'anticipazione nell'articolo di Bush.

4.2.5 Le funzionalità del Memex

La maggior parte dei contenuti del Memex è archiviata su microfilm (con rapid selector). Il microfilm è un grande limite del Memex, in quanto analogico.

Le potenzialità del collegare elementi:

- ⇒ Fornisce un passo verso un'*indicizzazione associativa*³;
- ⇒ Quando un utente crea un percorso, il Memex lo *nomina*, *inserisce il nome nel libro dei codici* e lo batte sulla tastiera;
- ⇒ In fondo a ogni elemento ci sono degli *spazi vuoti* per immettere dei codici e viene impostato un puntatore per indicarne uno per ciascun elemento;
- ⇒ L'utente preme un singolo tasto e gli elementi vengono uniti in modo permanente⁴.

³Qualsiasi elemento può immediatamente selezionarne automaticamente un altro

⁴Nei relativi spazi appare il codice.

Questo processo fa sì che, in qualsiasi momento, quando uno di questi elementi viene richiamato, venga richiamato anche l'altro⁵. Si può dire che gli oggetti fisici siano stati raccolti da fonti remote e rilegati insieme per *formare un nuovo libro*⁶.

Esempio 4.2.3 (Un Caso d'Uso del Memex)

Questo esempio è ripreso parola per parola dalle slides del prof. Cardone.

Poniamo il caso che il proprietario del memex sia interessato all'origine e alle proprietà dell'arco e delle frecce. Più precisamente egli sta studiando la ragione per cui l'arco corto turco fu apparentemente migliore dell'arco lungo inglese nei combattimenti delle Crociate.

Egli ha dozzine di libri e articoli potenzialmente pertinenti nel suo memex.

Prima sfoglia un'encyclopedia, trova un articolo interessante ma non abbastanza dettagliato, e lo lascia proiettato. Poi, in un libro di storia, trova un altro articolo pertinente e collega i due insieme. Procede in questo modo, creando un percorso composto da molti elementi.

Occasionalmente inserisce un proprio commento, collegandolo al percorso principale o, attraverso un percorso laterale, a uno specifico elemento.

Quando diventa evidente che le proprietà di elasticità dei materiali a disposizione avevano molto a che fare con l'arco, egli crea una ramificazione su un percorso laterale che lo porta a testi sull'elasticità e a tabelle di costanti fisiche.

Egli inserisce una pagina di analisi scritta a mano da lui stesso.

Quindi crea un percorso di suo interesse attraverso il labirinto dei materiali a sua disposizione. Diversi anni più tardi [con] un tocco accede al libro dei codici. Premendo alcuni tasti egli proietta l'inizio del percorso. Con una leva lo scorre a piacere fermandosi sugli elementi interessanti, facendo delle digressioni. Appariranno tipi totalmente nuovi di encyclopedie, già munite di una rete di tracce associative che le attraversano, pronte per essere immesse nel memex dove vengono ampliate. [Esempi sull'uso di una grande massa di informazioni da parte di avvocati, medici e scienziati.]

Lo storico con un vasto resoconto cronologico di un popolo lo accosta ad un percorso saltuario che si sofferma solamente sui temi salienti, e può seguire in ogni momento percorsi paralleli che lo portano a spaccati di civiltà in particolari epoche.

Meta-nota 4.2.2

Wikipedia è nata in un'ottica di collegamenti e di ipertesto. Il concetto di *wiki* è stato inventato da Ward Cunningham. Il wiki è ispirato agli ipertesti e fu implementato per la prima volta in *HyperCard*. Inoltre Cunningham è anche noto in ambito di programmazione OO per l'Extreme Programming (XP), visto nel corso "Sviluppo Applicazioni Software".

Definizione 4.2.6: Apripista

Un'apripista è una nuova professione che si occupa di stabilire percorsi utili attraverso l'enorme massa delle informazioni archiviate.

4.3 Il progetto di Doug Engelbart: Demo

- ⇒ 1945: Legge "As we may think" di Vannevar Bush;
- ⇒ 1957: Entra alla Stanford Research Institute (SRI);
- ⇒ 1962: Pubblica il rapporto "Augmenting Human Intellect: A Conceptual Framework";

⁵Proprio come un ipertesto.

⁶Quest'idea (un libro costruito attivamente da chi lo sta consultando), anticipata da Otlet, verrà ripresa da Licklider a proposito delle *biblioteche del futuro*.

- ⇒ 1963: partono i finanziamenti per il progetto allo "Augmentation Research Center" (ARC) ⁷;
- ⇒ 1968: La "madre di tutte le dimostrazioni" (Demo), San Francisco;
- ⇒ 1969: Primo collegamento ARPANET⁸ (tra UCLA e ARP).

Definizione 4.3.1: Augmenting Human Intellect

Augmenting Human Intellect (il rapporto del 1962) contiene la proposta del progetto e *anticipa* tutte le realizzazioni tecnologiche fino al 1968. Esso punta a migliorare l'efficacia intellettuale dell'uomo, scivolando nell'interazione/simbiosi tra uomo e macchina.

Meta-nota 4.3.1

A un certo punto del progetto si fece uso di uno strumento hardware a 5 dita e gli operatori comunicavano tra loro, da una postazione all'altra, mostrando le dita della mano.

4.3.1 L'idea di "Augmentation"

Engelbart puntava ad affrontare i problemi "*complessi*": di diplomatici, di scienziati, di fisici, avvocati, etc.. Questi problemi vanno affrontati non tramite trucchetti, ma comprendendo il problema. I problemi possono essere ben posti/tame problems o mal posti/wicked problems (sul lato della progettazione⁹).

wicked problems	tame problems
il problema non è compreso fino a quando non si sviluppa una soluzione	hanno una formulazione ben definita e stabile,
non c'è un punto in cui il problema possa dirsi chiuso	hanno un punto di chiusura definito, in cui la soluzione si può dire raggiunta,
le soluzioni non possono dirsi giuste o sbagliate	la soluzione, se esiste, può essere valutata oggettivamente come giusta o sbagliata,
ogni problema di questa classe è essenzialmente nuovo e unico	appartengono a classi di problemi risolubili tutti allo stesso modo,
nessuna ricerca di una soluzione può procedere per tentativi ed errori (non si può costruire un'autostrada per vedere come funziona)	hanno soluzioni che possono essere tentate o abbandonate facilmente,
non ci sono soluzioni alternative preassegnate	hanno insiemi limitati di soluzioni alternative.

Figure 4.2: Problemi mal posti vs. Problemi ben posti.

Note:-

La questione dei problemi mal posti e ben posti è affrontata approfonditamente nel corso "Metodologie e Tecnologie Didattiche per l'Informatica".

Esempio 4.3.1 (Quali wicked e quali tame?)

Con ✗ si indicano i wicked problems e con ✓ i tame problems.

⁷Anche grazie a Licklider.

⁸Uno dei primi nodi a essere associati ad ARPANET fu proprio quello del SRI.

⁹Un esempio è la pianificazione di una linea di metropolitana. I problemi urbanistici sono generalmente wicked problems.

- ✓ Trovare la radice quadrata di 7358;
- ✓ Trovare il cammino più breve tra un punto A e un punto B;
- ✗ Progettare un'automobile sicura;
- ✓ Riparare una lavatrice;
- ✗ Costruire una linea di metropolitana.
- ✓ Racimolare 10.000 euro;
- ✗ Organizzare una mostra;
- ✗ Preparare un disegno di legge.

Esempio 4.3.2 (Un architetto "aumentato")

Engelbart racconta l'esempio di un architetto che sta progettando un edificio.

Attività di questo architetto aumentato da un computer (il "*clerk*"):

- ⇒ Ha già fantasticato su diverse strutture e le *mette alla prova sullo schermo*;
- ⇒ Sullo schermo ha una *vista prospettica* del sito di costruzione sul pendio della collina sormontata dalla sede stradale, rappresentazione dei vari alberi che devono rimanere sul terreno e i vari punti di allacciamento per i servizi;
- ⇒ Con il *puntatore* indica due punti di interesse;
- ⇒ Dopo un po' l'architetto *cambia la scena* sullo schermo in una visione dall'alto che mostra lo scavo;
- ⇒ *Immette con la tastiera un elenco di elementi* controllandoli uno a uno mentre appaiono sullo schermo, rimandandone lo studio in seguito.

4.3.2 Classi di Strumenti di Aumentazione

- ⇒ *Artefatti*: oggetti fisici progettati per la comodità dell'uomo, per la manipolazione di cose, materiali e simboli;
- ⇒ *Linguaggio*: i modi in cui l'individuo segmenta l'immagine del mondo in concetti che la sua mente usa per modellare, i simboli che associa a quei concetti e utilizza nella manipolazione consapevole dei concetti (il "pensiero");
- ⇒ *Metodologia*: i metodi, le procedure e le strategie con cui un individuo organizza la sua attività finalizzata alla soluzione di problemi;
- ⇒ *Addestramento (training)*: il condizionamento necessario all'essere umano per portarlo a utilizzare i precedenti elementi in modo efficace.

Meta-nota 4.3.2

Questo sistema è: Trained Human being together with his Language, Artefacts and Methodology (H-LAM/T^a). Un esempio di un sistema H-LAM/T sono delle possibili "biblioteche del futuro" (Licklider) in cui sono presenti vari artefatti (scaffali, schedari, etc.), componente umana (bibliotecari, catalogatori, etc.), l'addestramento (dei bibliotecari). Il tutto ciò poteva essere sostituito da loro versioni digitali che "ri-media" dei media analogici in digitali.

^a"Purtroppo Engelbart usava degli acronimi assurdi", citazione necessaria del prof. Cardone

Meta-nota 4.3.3

Benjamin Lee Whorf, ingegnere chimico e poi linguista, affermava che il linguaggio influenza il pensiero. Whorf studiava le lingue dei nativi americani, insieme a Edward Sapir. Questi due linguisti formularono l'ipotesi per cui la concettualizzazione del mondo dipendesse dal linguaggio.

IN "Science and Linguistics" Whorf mostrò il modo di segmentare le esperienze di varie popolazioni a confronto con quello degli inglesi. Per esempio il fatto che gli eskimesi avessero diverse parole per "neve". Per ricollegarci al tema di questo corso, Whorf influenzò gli informatici come Nelson (che si vedrà più avanti) ed Engelbart (che lo cita direttamente nel suo lavoro).

4.3.3 Il Memex come strumento per la strutturazione simbolica

I percorsi associativi del Memex forniscono una capacità di strutturazione di simboli che deriva da una nuova capacità in termini di artefatti e processi¹⁰. Inoltre il Memex può aumentare le capacità umane di strutturazione ed esecuzione dei processi. Engelbart poneva un altro esempio: un sistema di *schede a bordi perforati [edge-notched]*, a funzionamento manuale. Le "registrazioni unitarie", a differenza di quelle del Memex, sono pezzi di testo o manoscritti su *schede di formato IBM a bordi perforati*. Esse sono dati, pensieri, fatti, considerazioni, concetti, idee, preoccupazioni, etc. pertinenti a un *determinato problema*.

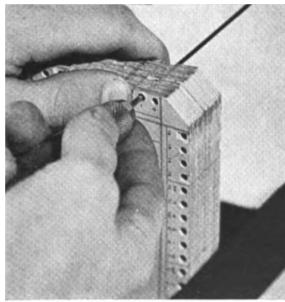


Figure 2-1. Start of single needle direct sort.

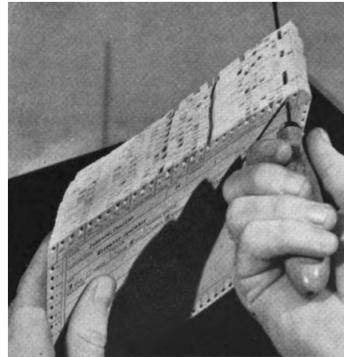


Figure 2-2. Second step in sorting cards.

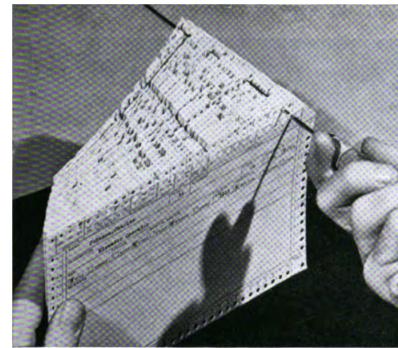


Figure 2-3. Third step—cards are fanned out.

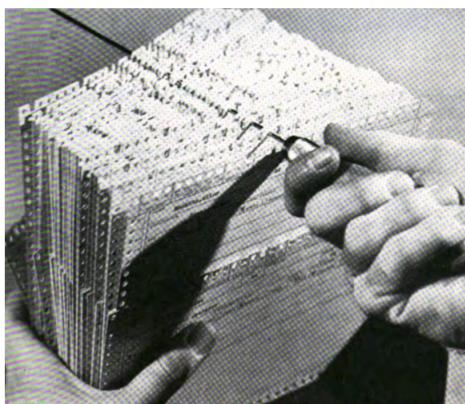


Figure 2-4. Fourth step—separation of cards begin.

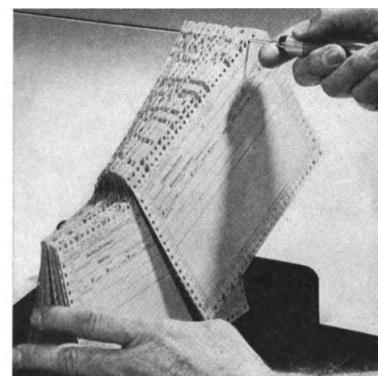


Figure 2-5. Fifth step—vertical edge on alignment block helps separate dropped cards.

Figure 4.3: Gestione delle schede perforate.

Note:-

Queste schede sono alla base di HyperCard, il primo sistema di ipertesto.

¹⁰Il Memex è un artefatto che supporta queste nuove strutture di concetti.

Meta-nota 4.3.4

La novità dell'utilizzo di queste "card" è l'utilizzo di unità minime di informazione, di insiemi di argomento ristretto che forniscono grande flessibilità e capacità di manipolazione. Esse mettono a disposizione uno *spazio di lavoro* che si può sfogliare^a, a cui si possono fare aggiunte o correzioni e che permettono la costruzione di nuovi insiemi di nuclei di pensieri.

^a"Browse".

4.3.4 Utilità del computer

- ⇒ Un computer è direttamente in grado di eseguire processi primitivi di manipolazione simbolica;
- ⇒ Con quei processi primitivi si possono costruire ogni altro processo che possa essere descritto da un linguaggio;
- ⇒ Un programma è una struttura di processi primitivi memorizzato sotto forma di simboli;
- ⇒ I computer hanno vari modi per memorizzare simboli in modo che possano essere manipolati;
- ⇒ È possibile descrivere al computer nuovi simboli;
- ⇒ L'interfaccia è il punto di contatto tra l'uomo e la macchina (nascita dell'interazione uomo-macchina¹¹);
- ⇒ 1962: *un programma sperimentale di ricerca che è ancora lontano alcuni anni dalla fase presente della sua realizzazione* (DEMO del 1968).

4.4 Panoramica della DEMO

Sezioni:

1. La più celebre introduzione dell'informatica, descrive, in tono epico, la situazione che il sistema nLS intende aumentare con l'uso del computer;
2. Manipolazione di testi, ri-organizzazione e visualizzazione sullo schermo del computer. Livelli e outline. Hyperlinks;
3. La metodologia di ricerca: l'approccio bootstrap (gruppo di ricerca) sul sistema nLS e sui principi di progettazione per lo sviluppo di sistemi di aumentazione;
4. Periferiche: il mouse e il "keyset" (tastiera);
5. Descrizione dell'hardware del sistema;
6. Software di sistema (con Jeff Rulifson);
7. Applicazioni del sistema: studio e modifica di articoli;
8. Collaborazione online tra due persone attraverso condivisione di file e dello schermo (con Bill Paxton);
9. Collegamento di computer in rete con condivisione di programmi ed esecuzione remota (viene annunciato di ARPANET);
10. Presentazione del gruppo di ricerca.

Meta-nota 4.4.1

Engelbart, inoltre, fu un sostenitore del lavoro di gruppo, infatti e nel suo gruppo di ricerca che nasce la figura del *facilitatore visuale*: una persona che durante le riunioni si occupa di disegnare su un foglio ciò che viene detto.

¹¹Importante il time-sharing

Tuttavia, il laboratorio di Engelbart fallì perché propose un'immagine dell'utente che non ebbe successo (l'idea di un utente di un personal computer che rimane circoscritto dal sistema). Molti dei collaboratori di Engelbart andarono a lavorare alla Xerox PARC che proponeva un'idea di utente diversa, più realistica.

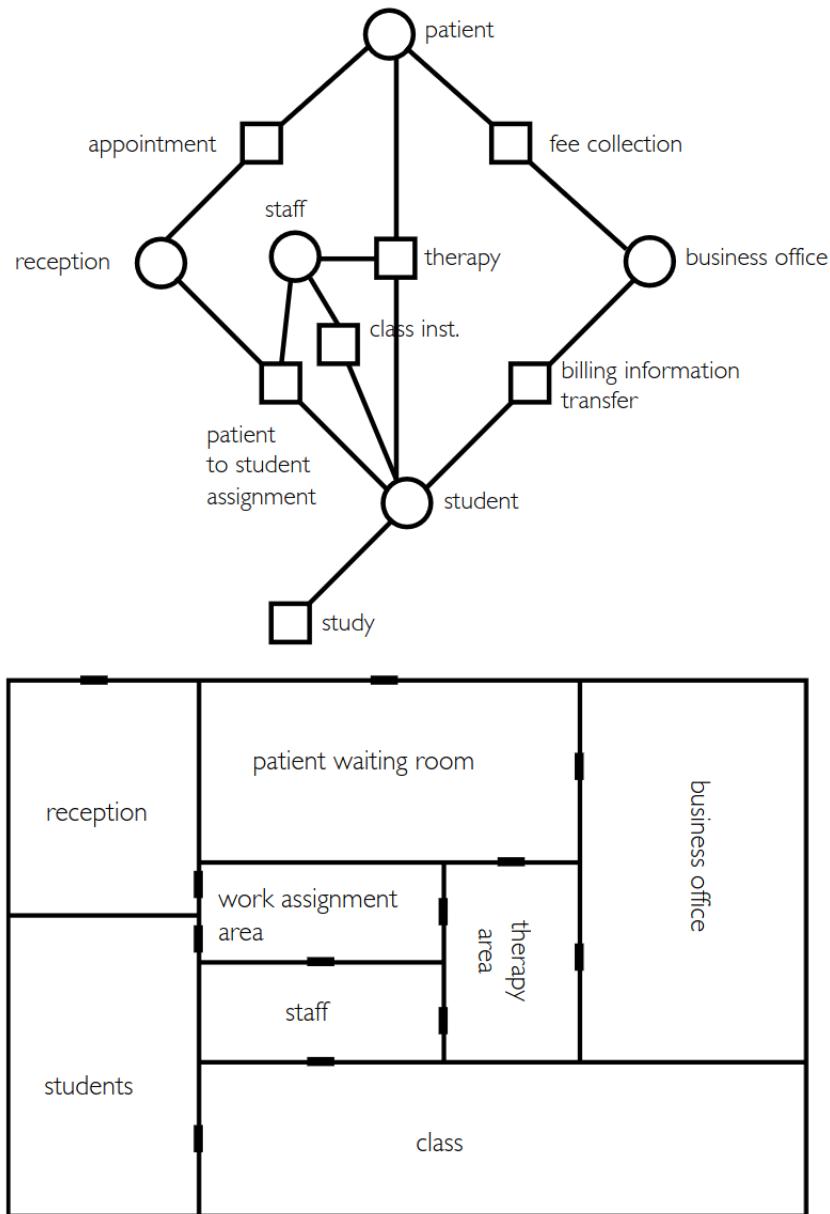


Figure 4.4: Il sistema di classificazione della scuola dentistica dell'Università di Boston.

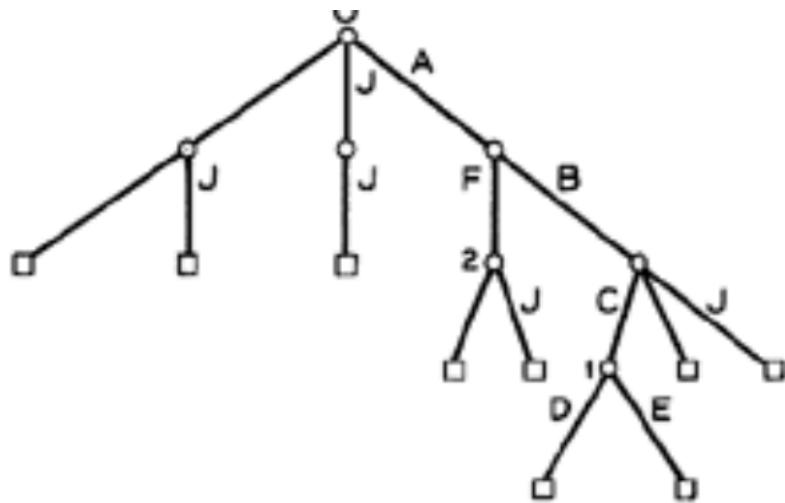


Figure 4.5: Filesystem UNIX.

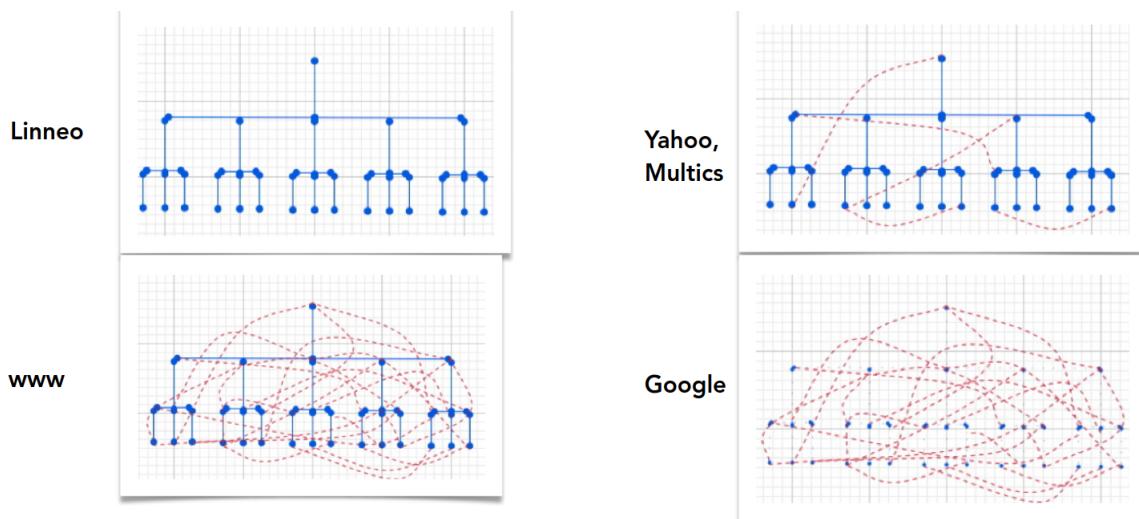


Figure 4.6: Evoluzione delle gerarchie di classificazione.

5

Ted Nelson e Licklider

6

Paul Otlet e Alan Kay

7

Letture

Questo capitolo contiene dei mini-riassunti delle Letture obbligatorie per l'esame. Poiché è possibile e consigliato consultare i documenti durante l'esame, questi riassunti hanno la funzione di aiutare a ricordare i concetti principali per facilitare la navigazione e la consultazione.

- ⇒ Vannevar Bush, Come potremmo pensare, 1945;
- ⇒ Susan Barnes, Douglas Carl Engelbart: Developing the Underlying Concepts for Contemporary Computing, 1997;
- ⇒ J. C. R. Licklider, Man-computer symbiosis, 1960;
- ⇒ Lev Manovich, Alan Kay's Universal Media Machine, 2013.

7.1 Come potremmo pensare

7.2 Douglas Carl Engelbart: Developing the Underlying Concepts for Contemporary Computing

7.3 Man-computer symbiosis

7.4 Alan Kay's Universal Media Machine

8

Domande

- 8.1 Lullo, Leibniz e la storia del calcolo
- 8.2 Turing e la fisica del calcolo
- 8.3 Funzioni calcolabili e combinatori
- 8.4 "As we may think"
- 8.5 Engelbart e Nelson
- 8.6 Otlet, Lickider e Kay