

Appunti di Storia dell'Informatica

AA 2022/2023

Indice

Copertina	1
Indice	2
Sinossi	3
Prologo	7
Glossario:	7
La materia e gli strumenti:	7
Dal mito alla filosofia	10
Sistemi di scrittura:	10
Sistemi di numerazione:	12
Sistemi formali:	13
L'evoluzione medioevale	15
Il sillogismo:	15
L'algebra in Europa:	16
Le rivoluzioni moderne	18
L'importanza della matematica:	18
Calcolo filosofico:	19
La nascita delle macchine:	20
La nascita dell'informatica	22
La rivoluzione industriale nel calcolo:	22
Sviluppo dei dispositivi linguistici:	23
Macchine calcolatrici:	26
La fusione delle discipline	32
Shannon:	32
Cibernetica:	33
La fusione di Von Neumann:	33
Il computer entra nella società	36
Hardware:	36
Software:	38
L'utente del computer:	38
Seminari	40
Progetto SAGE:	40
Progetto SABRE:	42
Progetto ERMA:	45
Domande da esame	47

Sinossi

La sinossi contiene problemi che hanno richiesto la creazione di soluzioni nuove, che quindi sono particolarmente significativi per la storia dell'informatica.

Al centro (Problem solving) si trova il problema da risolvere, mentre nelle colonne laterali si trovano i dispositivi linguistici (se esistenti) ed operativi utilizzati o inventati per risolverlo.

Legenda:

- *Corsivo*: elementi la cui classificazione presenta delle criticità.
- **Grassetto e sottolineato**: tappe più importanti.

Dispositivi linguistici	Problem solving	Dispositivi operativi
<i>Codifica dell'informazione: fisica, chimica, biologia, ...</i>	<i>Comprensione degli eventi naturali</i>	<i>Universo capace di codificare l'informazione</i>
Lingue naturali	Comunicare	Homo sapiens
Sistemi di scrittura	Descrivere e Raccontare	Homo sapiens
Sistemi di numerazione	Organizzare e Contare	Homo sapiens
Traccia di conteggi su tavoletta	Gestione di magazzino	Contabili
Tavolette Biblioteca Ebla	Contabilità commerciale	Commercianti assiro-babilonesi
I testi delle leggi di Hammurabi	Amministrazione della giustizia	Giudici di Hammurabi
Papiro di Ahmes	Sistema scolastico	Docenti e studenti (con l'aiuto dell'abaco)
Grammatica del sanscrito	Produzioni linguistiche corrette	Parlanti il sanscrito
I Ching	Produzione di oracoli	Sacerdoti cinesi
"Tutto è numero"	Filosofia, esigenza di capire il mondo	Pitagorici
Dialettica e retorica	Ricerca della verità e strategie per convincere	Filosofi, giuristi e politici
I dialoghi di Platone	Dialettica e retorica	Filosofi
Sillogismo di Aristotele	Argomentare in modo cogente	Filosofi
Sistema <i>formale</i> di Euclide	Costruzioni in Geometria e dimostrazioni	Filosofi e matematici

Matematica, fisica e astronomia	<i>Calcoli geometrici, idraulici ed astronomici</i>	<i>Erodoto, Archimede, Tolomeo</i>
	<i>Misurare il tempo</i>	<i>Orologi a acqua e clessidre (macchine analogiche)</i>
Aritmetica e algebra	Calcolare	Matematici indo-arabi
Sillogismo medievale	Logica	Filosofi
<i>Liber abaci</i>	In Europa: dall'abaco all'algebra indo-araba	Fibonacci ed anche "abacisti e algebristi"
Trigonometria e logaritmi	Calcoli astronomici, geodetici e marittimi	<i>Copernico, Colombo, Ticho Brahe, Keplero</i>
	<i>Misurare il tempo</i>	<i>Orologi meccanici (macchine digitali)</i>
Tavole numeriche	Facilitare il calcolo	"Naviganti" ed astronomi
Ars Magna combinatoria	Convincere convertire	Lullo
Linguaggio naturale certificato da filosofi	Argomentazione naturale come calcolo	Hobbes
Linguaggio artificiale <i>Quo facto [...] calculemus</i>	Argomentazione formale come calcolo	Leibniz
Enciclopedia delle arti e dei mestieri (pseudoalgoritmi)	Descrivere ed insegnare le professioni	Diderot e D'Alembert
Metodo di Smith	Scomporre il lavoro	Professionisti e (successivamente) macchine
Schede perforate	Giochi e passatempi	Automi
Schede perforate	Produrre tessuti	Telaio di Jacquard
Calcolo numerico e differenze finite	Tabulare polinomi per approssimare funzioni	Matematici
de Prony: scomposizione del lavoro matematico	Costruire tavole numeriche in maniera efficiente	Matematici tecnici e contabili (<i>computer-1</i>)
	Automazione del lavoro dei contabili	Muller
	Produzione di tavole numeriche senza interventi umani	Macchina alle differenze di Babbage e " <i>computer-2</i> "
Linguaggio di Menabrea e Ada Lovelace	Calcolo di espressioni aritmetiche	Macchina analitica di Babbage

	<i>Utilizzo dell'elettromeccanica</i>	Telai, calcolatrici, telefono, telegrafo
	Elaborazione dati di censimenti	Selezionatrice di Hollerith usata da impiegati
	Meccanizzazione aziendale	Macchine di Hollerith IBM serie 600
Plankalkül	Calcolo digitale automatico	Progetto di Zuse
"Protoprogrammazione"	Calcoli a supporto della WW2 negli USA	Macchine elettromeccaniche
"Costruttori" di programmi	Calcoli balistici	L'elettronica in ACE, Mark I ed ENIAC
Algebra di Boole	Calcolo proposizionale e manipolazione di insiemi	Matematici
Sistema formale di Frege e Russell	Logica matematica e logicismo	Matematici
<i>Macchina astratta</i> (sarà implementata nel 1956)	Dedurre l'aritmetica	Russell e Whitehead
<i>Macchina astratta</i>	Dedurre la geometria	Hilbert
	Problemi di Hilbert	
Teoremi di Gödel	Fine del logicismo	
Macchine di Turing (programmi)	<u>Definizione di algoritmo</u>	"Computer"
<i>Macchina</i> Universale di Turing (astratta)	<u>Definire la calcolabilità</u>	Matematici
Neuroni	Progetto CIBERNETICA	L'elettronica
Algebra di Boole e Circuiti elettronici	Dall'analogico al digitale	Shannon
Macchina universale di von Neumann (astratta)	<u>Nascita dell'Informatica</u>	Macchina universale di von Neumann: <u>il Computer</u>
Progetto IAS	Moore School Lectures	
	Informatica e mercato	Il progetto UNIVAC
	Informatica e mercato	Il progetto IBM
<i>Real time</i>	Progetto SAGE	

	Progetti SABRE ed ERMA	
Linguaggi di programmazione	<i>Good programming</i>	Informatici
Software di base	Centri di calcolo Gestione ottimale dello Hardware	Industrie Informatica
Software applicativo	Facilitare e standardizzare la soluzione di problemi	Industrie Informatica <i>software house</i>
Time-sharing	Partecipazione diretta degli utenti	Reti di terminali
Linguaggi per IA	Simulazione dell'intelligenza	Informatica quantistica

Prologo

Glossario:

- **Calcolo:** tutto ciò che può essere fatto da un dispositivo operativo seguendo le indicazioni del dispositivo linguistico.
- **Digitale:** che può essere espresso tramite un insieme finito di simboli.
- **Effettivo:** di un linguaggio, che descrive la gestione dell'informazione come manipolazione di simboli, usa un numero limitato di parole chiave non ambigue ed ogni sua frase grammaticalmente corretta ha uno ed un solo significato, mentre costruzioni sgrammaticate non hanno alcun significato.
- **Automatico:** che è in grado di eseguire in maniera autonoma manipolazioni di simboli opportunamente descritte con linguaggi formali.
- **Problem posing:** capacità di definire le specifiche di un problema.
- **Problem solving:** capacità di elaborare strategie per risolvere un problema.
- **Calcolabilità:** possibilità di calcolare la soluzione di un problema.
- **Trattabilità:** possibilità di calcolare la soluzione di un problema impiegando una quantità di risorse ragionevole.
- **Cogente:** affermazione che non ammette dubbi.
- **Esattamento** (exaptation in inglese): una struttura, nata con una funzione, si evolve ed acquisisce una funzionalità nuova (per esempio: le piume nascono con lo scopo di proteggere il corpo e poi diventano fondamentali per il volo, oppure i computer nascono come macchine per calcolare tavole e poi diventano general purpose).

La materia e gli strumenti:

L'informatica nasce come ricerca di un linguaggio che non abbia le stesse irregolarità (ed ambiguità) del linguaggio naturale.

Tale linguaggio viene poi utilizzato nelle più disparate discipline che necessitano di formalismi non ambigui.

Di conseguenza la **storia dell'informatica** è anche la **storia del problem solving** in moltissime discipline.

Per studiare la storia dell'informatica si deve quindi evidenziare di ogni fenomeno come è nato, come sono stati trovati gli strumenti per risolverlo, quali influenze hanno contribuito alla sua creazione e quali ambiti ha a sua volta influenzato.

Per parlare di questi linguaggi non ambigui è però necessario utilizzare un **metalinguaggio** comprensibile a coloro che studiano la materia, è quindi necessario dare nuove definizioni ed applicare delle restrizioni al linguaggio naturale per limitare l'ambiguità nel trattare gli argomenti del corso.

Lo scopo di questa sezione (e del glossario precedente) è definire gli strumenti linguistici che si utilizzeranno nel resto del corso.

I dispositivi che vengono studiati possono essere classificati in varia maniera in base alle loro caratteristiche:

Per quanto riguarda la loro **natura** possono essere:

- **linguistici**: per la descrizione dei problemi e delle relative soluzioni
esempio: i linguaggi di programmazione
- **operativi**: per eseguire i procedimenti risolutivi
esempio: un manufatto, come il computer stesso

Attenzione: le macchine virtuali sono dispositivi linguistici.

Rispetto invece alla loro **espressività** si possono classificare come:

- **universali**: possono esprimere qualunque cosa
esempio: i linguaggi naturali o un computer moderno
- **generali**: possono esprimere molte cose relative ad uno specifico ambito
esempio: il testo delle leggi da Hammurabi o una lavatrice
- **specifici** (o speciali): possono esprimere una cosa sola
esempio: un singolo programma o un telecomando.

Per la loro **struttura** si possono dividere in:

- **formali** (o artificiali): dispositivi effettivi, definiti in maniera non ambigua da un insieme di regole in tutto il loro funzionamento
- **non formali** (o naturali): non definibili in maniera precisa

Riassumendo con un esempio si ha:

Dispositivi operativi	Formali artificiali	Non formali (pseudo)naturali	Dispositivi linguistici	Formali artificiali	Non formali (pseudo)naturali
universali	computer	<i>Homo sapiens</i>	universali	Linguaggi dei computer	Linguaggi Naturali
generali	Elettrodomestici programmabili	giudici di Hammurabi	generali	Sql, RPG, word	Testi legislativi Ricettari, protocolli
speciali	Elettrodomestici non program.li	<i>Chaplin</i> in tempi moderni	speciali	Programmi algoritmi	Una ricetta di cucina

Inoltre tutti i dispositivi linguistici devono avere tre proprietà (?):

- effettivi
- efficaci
- agevoli

Tutta l'evoluzione dell'informatica può essere divisa in due fasi:

- **fase storica**: dall'inizio della storia all'ingresso del computer in società (1950 circa)
- **fase contemporanea**: dalla diffusione del computer ai giorni nostri.

La storia dell'informatica deve essere vista sotto tre punti di vista distinti (nel corso verranno trattati tutti e tre):

- **linguistico-formale**: l'evoluzione dei linguaggi sia dal punto di vista storico che guardando alla struttura sempre più complessa che assumono.
- **strumentale-operativo**: l'evoluzione dei dispositivi operativi, ovvero degli ausili strumentali utilizzati per facilitare o realizzare interamente il trattamento, la trasmissione e la memorizzazione delle informazioni.

- **applicativo-metodologico**: l'evoluzione del pensiero computazionale e delle esigenze che hanno portato alla nascita dell'informatica (e che l'informatica deve risolvere).

Dal mito alla filosofia

Dall'inizio della storia dell'uomo è stato innanzitutto necessario avere degli strumenti per comprendere il mondo circostante e per comunicare, a causa di ciò nascono i linguaggi naturali.

Successivamente (non si è in grado di stabilire cosa sia avvenuto prima) sorge la necessità di scrivere e contare, di conseguenza nascono i sistemi di scrittura e numerazione.

Nell'arco di alcuni millenni si passa quindi da una cultura orale ad una scritta; questo passaggio avviene con lo sviluppo di tre ambiti fondamentali:

- **sistemi di scrittura**: per descrivere e raccontare,
- **sistemi di numerazione**: per contare,
- **sistemi formali**: per argomentare e dimostrare.

Sistemi di scrittura:

40000 anni fa compaiono le prime tracce rupestri, ma solo nel 3000 a.C. con i Sumeri nasce quello che è definibile un sistema di scrittura (a pittogrammi).

Ogni comunità **sviluppa** il proprio **sistema di scrittura** e numerazione in maniera sufficientemente **autonoma**, influenzandosi poco a vicenda e con scopi molto diversi tra loro.

Di conseguenza si formano **quattro principali aree** in cui si sviluppa la scrittura in modo autonomo:

- Mesopotamia,
- Egitto,
- India,
- Cina.

Mesopotamia:

In mesopotamia la scrittura nasce con scopi amministrativi per la gestione del commercio e della contabilità.

Questa scrittura ha inizialmente forma pittografica.

Un esempio notevole di ciò è **la tavoletta sumerica del sito di Fara** (Šuruppak in sumero) in Iraq del 2650 a.C. circa; questa tavoletta descrive una distribuzione di orzo e mostra come era fatta una divisione, è quindi un documento necessario per l'amministrazione di un magazzino.

Sono necessari parecchi secoli affinché la scrittura evolva e venga utilizzata anche per altri scopi, un primo esempio di ciò è un'iscrizione funeraria del 2700/2600 a.C. che riporta il nome ed il titolo del defunto.

Nel tempo i pittogrammi vengono sostituiti dal cuneiforme, un esempio di questo nuovo metodo di scrittura è il poema che descrive le gesta di Gilgamesh del 2700/2500 a.C.

Nelle **tavolette della biblioteca di Ebla** si hanno invece delle tavole su cui si trovano informazioni specifiche sugli oggetti degli scambi; tali tavolette sono ordinate per tema su scaffali e la prima tavoletta di ogni scaffale è la descrizione del contenuto.

Questo è un primo tentativo di costruire un indice per permettere di trovare facilmente ciò che si cerca in un insieme di testi.

Nei testi delle **leggi di Hammurabi** (1750 a.C.) c'è un primo tentativo di formalizzare un insieme di leggi tramite casi, anticipa quindi la struttura condizionale (if then else) dei sistemi basati su regole (e anche lontanamente dei sistemi esperti, una branca dell'intelligenza artificiale che simula il comportamento di uno o più agenti esperti in un campo).

Un'altra cosa fondamentale per la storia dell'informatica è che con questa opera Hammurabi tenta di produrre un insieme di sentenze (282 per la precisione) che siano effettive, ovvero che non possono essere interpretate in maniera ambigua e che per ogni controversia dello stesso tipo venga formulata la stessa sentenza.

Le leggi di Hammurabi sono quindi un dispositivo linguistico generale (tratta ogni tipo di lite giudiziaria) non formale (scritto in linguaggio naturale anche se disciplinato), i suoi dispositivi operativi sono i giudici.

Egitto:

A partire dalla seconda metà del 4° millennio a.C. nell'antico Egitto si sviluppa un tipo di scrittura pittografica utilizzata per le iscrizioni funerarie, sacre e solenni nei templi e sulle tombe.

Questa scrittura poi si evolve nel sistema geroglifico, dove i simboli rappresentano sia oggetti che suoni (più o meno come accade nei rebus).

Nel 600 a.C. compaiono documenti come il **Papiro di Ahmes**, ovvero documenti a scopo didattico che illustrano la risoluzione di vari problemi di tipo matematico, quindi inizia la ricerca di un linguaggio formale, ma non viene sviluppato e le indicazioni vengono date tramite un insieme di esempi.

India:

In India la scrittura nasce legata al culto religioso: nella seconda metà del 2° millennio a.C. viene adottata una forma scritta del sanscrito per stendere i Veda, ovvero i testi sacri del vedismo (da cui successivamente si sviluppa l'induismo) scritti tra il 1500 ed il 1200 a.C.

Un passo importante per la storia dei sistemi di scrittura è la codifica della **grammatica del sanscrito** operata da **Panini** tra il 6° ed il 4° secolo a.C.

Tale grammatica è il primo esempio di un dispositivo linguistico universale non formale, che contiene le regole per produrre tutte le frasi del linguaggio naturale (i dispositivi operativi sono tutte le persone che vogliono parlare il sanscrito).

Il metodo con cui viene descritto il sanscrito ha alcune similitudini con i metodi moderni utilizzati per scrivere un compilatore.

Cina:

In Cina tra 1200 ed 800 a.C. si evolve un metodo di scrittura di tipo ideografico come strumento divinatorio.

Le testimonianze più antiche sono state trovate a Jiahu, un sito neolitico sul fiume Huai nella provincia dello Henan, dove sono stati rinvenuti dei gusci di tartaruga del 6500 a.C. sui quali erano stati incisi diversi simboli.

L'invenzione dei caratteri ideografici cinesi viene tradizionalmente attribuita a Cangjie (2650 a.C. circa), un funzionario del leggendario imperatore Huangdi.

In realtà si sa che il processo è stato ovviamente molto più graduale e si hanno reperti di ossa su cui sono incisi responsi oracolari e cocci (1600 a.C.) sui quali si vede un sistema di

scrittura completamente sviluppato che ha molti (ma non tutti) caratteri in comune col cinese moderno.

Si sviluppa un sistema formale per esprimere ed interpretare il risultato delle divinazioni descritto nel libro “**I Ching**” (il libro dei mutamenti).

La scrittura di questo libro è tradizionalmente attribuita al re fondatore della dinastia Zhou ed a Zhou Gong (morto circa nel 1032 a.C.); anche se probabilmente è effettivamente presente materiale dell'epoca, il libro è stato pesantemente modificato successivamente.

Questo sistema si basava inizialmente su una cifra binaria (si/no) alla quale veniva attribuita la risposta ad una

domanda, poi per raffinare il responso sono state aggiunte via via sempre più cifre fino ad arrivare agli esagrammi, ovvero 6 segni che possono assumere valore binario, quindi in totale si possono ottenere 64 diversi valori come risultato dell'attività divinatoria.

L'esagramma viene poi letto guardando le due coppie di tre “bit” che si formano (trigrammi), la trasformazione da esagramma a pronostico è un primo esempio di algoritmo rigoroso.

L'I Ching è un dispositivo linguistico non formale generale.

Superiore → Inferiore ↓	Qian Cielo	Zhen Tuono	Kan Acqua	Gen Montagna	Kun Terra	Xun Vento	Li Fuoco	Dui Lago
Qian Cielo	01	34	05	26	11	09	14	43
Zhen Tuono	25	51	03	27	24	42	21	17
Kan Acqua	06	40	29	04	07	59	64	47
Gen Montagna	33	62	39	52	15	53	56	31
Kun Terra	12	16	08	23	02	20	35	45
Xun Vento	44	32	48	18	46	57	50	28
Li Fuoco	13	55	63	22	36	37	30	49
Dui Lago	10	54	60	41	19	61	38	58

(Mediterraneo):

L'area mediterranea è marginale a questi fenomeni, ha però come forza quella di essere il punto d'incontro di tutte queste culture.

Alcune civiltà (in particolare i **fenici**) sviluppano un **sistema di scrittura alfabetico** (tramite **fonemi**, ovvero un simbolo indica un suono e non più un concetto) per semplificare la documentazione degli scambi tra popolazioni di lingue diverse, quindi un sistema per trascrivere le parole di lingue diverse.

Sistemi di numerazione:

Per quanto riguarda invece i sistemi di numerazione nascono nell'ambito del commercio e della gestione dei magazzini.

Inizialmente per tenere traccia di quanti beni si utilizzano **statuette di terracotta** rappresentanti i beni stessi, il numero di statuette che si possiede ne indica il numero.

Poi si passa al sistema delle **bulle**, ovvero scatole con all'esterno dipinto l'oggetto che si vuole contare ed all'interno un numero di sassolini pari al numero degli oggetti; questo è un passaggio molto importante in quanto nasce il concetto di numero come cosa separata dal concetto di oggetto da contare (quindi esiste il 3 in quanto numero, non solo il concetto di 3 mucche).

Si passa poi alle **tavolette** con il pittogramma dell'oggetto contato ed incise delle tacche per indicarne il numero.

Gli egizi sono i primi a creare dei **segni ad hoc** per rappresentare unità (barre verticali), decine (cerchi), centinaia (corde arrotolate), migliaia (fiori di loto), decine di migliaia (dita), centinaia di migliaia (girini o uccelli) ed i milioni (figura umana a braccia aperte) in un **sistema di numerazione additivo** (come quello romano).

Inoltre sviluppano anche un sistema di numerazione in base 12 per contare le ore del giorno, in cui ogni ora (ogni numero) è rappresentata da una falange di un dito della mano (la mano destra rappresentava il giorno e la sinistra la notte, in una mano se si esclude il pollice che veniva utilizzato per puntare ci sono 12 falangi).

Il **papiro di ahmes** è il primo esempio di libro scolastico che indica come risolvere dei problemi matematici tramite esempi (linguaggio non formalizzato), è un dispositivo linguistico formale speciale.

I babilonesi invece sviluppano un **sistema di numerazione posizionale** (come i numeri arabi) in base 60, probabilmente perché è facilmente divisibile per molti numeri.

Il primo numero a destra rappresenta le unità, quello alla sua sinistra le sessantine e così via con potenze crescenti di 60.

Le singole cifre (numeri da 1 a 60) però non hanno tutti simboli dedicati, ma vengono costruite in maniera additiva.

Traccia di questo sistema rimane probabilmente nel nostro modo di contare i minuti ed i secondi.

Come strumento per fare i calcoli viene inventato l'**abaco**, che per sua natura può funzionare in maniera efficace soltanto con sistemi di numerazione posizionali.

Non esiste ancora il concetto di 0, viene invece codificato molto bene il concetto di frazione e viene ampiamente sviluppata l'area dei calcoli approssimati introducendo all'utilizzo di tavole numeriche.

Sistemi formali:

In Grecia dal 500 a.C. si forma a pieno titolo la figura del filosofo ed il concetto di **filosofia**, in questo periodo inizia quindi a svilupparsi una ricerca del metodo con cui parlare dei problemi e di ciò che è vero in maniera rigorosa.

Per fare ciò all'interno della filosofia si sviluppano tre branche che studiano come trattare le argomentazioni in maniera diversa: la dialettica, la retorica e la logica:

- **dialettica**: discussione in linguaggio naturale tra persone alla pari disposte a modificare i propri pregiudizi, dove nel tempo si arriva ad una soluzione comune;
- **retorica**: una persona che conosce meglio di altri l'argomento parla in linguaggio naturale per convincere il gruppo interlocutore della verità della tesi che sta sostenendo;
- **logica**: vengono definite delle regole per ragionare e dei presupposti da cui partire (assiomi), poi si procede per inferenze (con regole di ragionamento comuni) fino a giungere ad una conclusione (in epoca medievale questa cosa viene poi espressa in linguaggio formale), lo strumento principale della logica è il **sillogismo**.

La differenza tra retorica e logica è poca, in quanto anche il retore utilizza delle regole per mostrare che ciò che si dice è vero, però la logica è molto più rigorosa ed efficace (è l'unica che assicura argomentazioni cogenti).

Nella ricerca del sapere oggettivo si iniziano anche a produrre strumenti e macchine che possano aiutare l'uomo nei suoi calcoli e ragionamenti, un esempio di ciò è la **macchina anticitera** (detta così perchè ritrovata davanti all'isola di Citera) che è una sorta di orologio astronomico del sistema solare conosciuto al tempo di Archimede ed è un precursore dei calcolatori analogici.

Per quanto riguarda il processo che ha portato alla nascita della logica, un primo tentativo di formalizzazione si trova nella celebre frase "**tutto è numero**" dei pitagorici, oggi si sa che questa frase è falsamente attribuita (apocrifa), ma è un buon indice di quella che è la loro mentalità (la quale pone le basi per la codifica digitale delle informazioni).

Socrate e Platone mettono fortemente in evidenza le debolezze del linguaggio naturale; Platone introduce come linguaggio per parlare del vero un linguaggio naturale disciplinato ed integrato con elementi più formali, mentre Socrate ritiene che non sia possibile trovare un linguaggio ma che la trasmissione del sapere si possa ottenere solo con la dialettica (motivo per cui non ha lasciato nulla di scritto).

Euclide invece formalizza il linguaggio della geometria (inventando la **geometria euclidea**, ma una nozione meno formale di geometria nel mediterraneo esisteva già in quanto sviluppata per la gestione dell'agricoltura dopo le piene del Nilo in Egitto) definendo in maniera formale gli strumenti utilizzabili e deducendo da essi tutto il resto con teoremi. Questa cosa è una rivoluzione perché viene inventata per la prima volta una lingua ad hoc per descrivere il mondo ed i ragionamenti in maniera formale.

Da ciò vengono sviluppati molti teoremi con cui interpretare il mondo in chiave geometrica (come i teoremi di Talete) e che permettono importanti calcoli, come il calcolo del raggio della terra di Eratostene.

Dalla geometria euclidea però si osserva l'esistenza di problemi non risolvibili (come i problemi non calcolabili in informatica), per esempio dividere un angolo in tre parti uguali (si dimostrerà solo 2200 anni dopo che non è risolvibile nella geometria euclidea), quindi c'è un forte impulso verso lo sviluppo di strumenti formali più potenti nel tentativo di risolvere problemi sempre più complessi.

La geometria euclidea è un dispositivo linguistico formale generale, che si poggia sui matematici come dispositivo operativo.

Come strumento che cerca di formalizzare il linguaggio naturale senza utilizzare la geometria **Aristotele** propone il **sillogismo**.

Questa idea viene poi elaborata in tutto il periodo medioevale fino a rendere i sillogismi un vero e proprio dispositivo linguistico formale (quello proposto da Aristotele è non formale).

Dalla logica e dalla aritmetica nasce quindi la capacità di **manipolare idee in maniera formale tramite l'uso di segni**, elemento fondamentale per lo sviluppo dell'informatica.

L'evoluzione medioevale

Il sillogismo:

Il sillogismo proposto da Aristotele viene elaborato nel medioevo fino a diventare un **dispositivo linguistico formale speciale** (le dimostrazioni fatte con i sillogismi sono molto limitate) per costruire argomentazioni cogenti.

I logici medioevali riformulano il sillogismo come un insieme di **24 formule** (programmi) per realizzare inferenze valide.

Per individuare le 24 deduzione valide tra tutte le 256 possibili però non è stato utilizzato un metodo formale, ma la loro **correttezza** è stata giustificata attribuendogli un **significato in linguaggio naturale** e verificando che il ragionamento abbia senso.

In epoca medioevale il sillogismo assume la seguente forma.

Il sillogismo è composto da due frasi dette **premesse**, di cui una **maggiore** e l'altra **minore**, ed una **conclusione**.

Premesse e conclusione sono **proposizioni dichiarative**, ciascuna delle quali contenente due **termini**: soggetto e predicato (da non intendere in senso grammaticale, infatti sia soggetto che predicato sono porzioni di frase che contengono al loro interno anche eventuali complementi).

Il termine presente nelle due premesse ma assente nella conclusione è detto **termine medio**.

Esempio:

Tutti gli uomini sono mortali (premessa maggiore)

Tutti i professori sono uomini (premessa minore)

Tutti i professori sono mortali (conclusione)

Ogni sillogismo valido può assumere una di queste **forme** (indicando con PM la premessa maggiore, Pm la premessa minore, C la conclusione, T1 e T2 due termini ed M il termine medio):

- $PM(T1, M) + Pm(T2, M) \Rightarrow C(T2, T1)$
- $PM(T1, M) + Pm(M, T2) \Rightarrow C(T2, T1)$
- $PM(M, T1) + Pm(T2, M) \Rightarrow C(T2, T1)$
- $PM(M, T1) + Pm(M, T2) \Rightarrow C(T2, T1)$

Le proposizioni dichiarative utilizzate nelle premesse e nelle conclusioni possono essere di 4 tipologie (dette **figure**) indicate nella simbologia medioevale dalle lettere A, E, I, O:

Simbolo	Figura	Esempio
A	Universale affermativa	Ogni S è P
E	Universale negativa	Ogni S non è P
I	Particolare affermativa	Qualche S è P
O	Particolare negativa	Qualche S non è P

Esempio:

Tutti gli uomini (M) sono mortali (T1): $A(M, T1)$

Tutti i professori (T2) sono uomini (M): $A(T2, M)$

Tutti i professori (T2) sono mortali (T1): $A(T2, T1)$

Questo sillogismo di tipo AAA è un'istanza particolare è della forma:

$A(M, T1) + A(T2, M) \Rightarrow A(T2, T1)$

Ovvero la premessa maggiore è di tipo A, la premessa minore è di tipo A e conclusione è sempre di tipo A.

Quindi in ognuna delle 4 forme è possibile combinare in varia maniera le 4 figure.

Nello specifico per ogni forma si possono avere $4 \times 4 \times 4 = 64$ combinazioni (4 premesse maggiori, 4 premesse minori e 4 conclusioni).

Inoltre siccome le forme sono 4 si hanno $64 \times 4 = 256$ tipi di sillogismo possibili.

L'elaborazione medievale ha mostrato che solo **24 di essi sono validi** (6 per ciascuna delle 4 forme), per fare ciò però non viene utilizzato un procedimento formale ma viene verificato per ogni tipo di sillogismo se esso ha senso in linguaggio naturale, di conseguenza il sistema dei sillogismi è ancora **dipendente dal linguaggio naturale**.

Tabella delle sigle dei 24 sillogismi validi:

$PM(T1, M) + Pm(T2, M) \Rightarrow C(T2, T1)$	$PM(T1, M) + Pm(M, T2) \Rightarrow C(T2, T1)$	$PM(M, T1) + Pm(T2, M) \Rightarrow C(T2, T1)$	$PM(M, T1) + Pm(M, T2) \Rightarrow C(T2, T1)$
AAA	EAE	AAI	AAI
EAE	AEE	IAI	AEE
AII	EIO	AII	IAI
EIO	AOO	EAO	EAO
AAI	EAO	OAo	EIO
EAO	AEO	EIO	AEO

L'algebra in Europa:

Nel medioevo l'algebra viene portata dall'India (dove si sviluppa enormemente) all'Europa attraverso gli arabi, introducendo il **sistema di numerazione posizionale**, il concetto di **zero** (inventato dall'indiano **Brahmagupta** nel 7° secolo d.C., anche se al problema della divisione per zero si arriverà successivamente) ed i primi elementi di aritmetica ed algebra. Si noti in particolare l'opera di **Muhammad al Khwarizmi** (da cui deriva la parola algoritmo) che intorno all'820 d.C. descrive in arabo le tecniche di calcolo indiane e la cui opera verrà tradotta in Europa.

L'algebra indo-araba è un dispositivo linguistico generale formale.

Intorno all'anno 1000 il monaco Gerberto (poi noto col nome di papa Silvestro 2°) tenta senza riuscire di diffondere in Europa l'utilizzo delle cifre decimali.

Nel 1120 Abelardo da Bath traduce in latino parte delle opere di Euclide e di Al Khwarizmi, iniziando quindi a farle circolare in Europa.

Fibonacci (il cui vero nome è Leonardo da Pisa, nato nel 1170) è un matematico che grazie al padre, il quale è un importante commerciante con un ufficio doganale in Algeria, è in grado di viaggiare molto in Egitto, Siria, Provenza, Grecia e Sicilia.

In questo modo entra in contatto con i matematici locali ed accumula un'ottima conoscenza della matematica indo-araba.

Inoltre utilizza come base di partenza per i propri studi gli Elementi di Euclide.

Nel 1202 scrive il **Liber Abaci**, dove con abaco si intende l'aritmetica e l'algebra (notazione utilizzata fino al 17° secolo).

Questa opera, che tratta della numerazione posizionale indiana, dell'utilizzo dello zero, delle operazioni con le frazioni, delle equazioni e di altri argomenti, ha grande diffusione in Europa e viene utilizzato come testo di riferimento per tre secoli.

Le novità introdotte però incontrano resistenza nel pubblico ed impiegano parecchio tempo a diffondersi, un esempio notevole di ciò è il divieto del 1280 ai banchieri da parte della città di Firenze di utilizzare le cifre arabe, questo perché si pensava che lo zero generasse confusione, favorisse le truffe e che in generale quel tipo di notazione potesse essere utilizzata per mandare messaggi segreti (da cui nasce il termine "cifrare").

Le rivoluzioni moderne

L'importanza della matematica:

Da questo momento lo studio della matematica si sviluppa fortemente in Europa ed in particolare in Italia.

Nel 1500 vengono scoperti metodi per calcolare le soluzioni di equazioni di terzo e quarto grado grazie al lavoro di Scipione del Ferro, Lodovico Ferrari, Pacioli e Cardano.

Viene anche sviluppato il calcolo letterale da Francois Viète.

Nel 1600 Nepero e Briggs introducono il concetto di logaritmo.

Per quanto riguarda la diffusione della numerazione decimale nel 1582 Simon Stevin, un matematico fiammingo, introduce la seguente notazione: 65,74 è scritto come 65⑦④②.

Nel 1592 lo svizzero Jost Bürgi semplifica la notazione di Stevin lasciando solo il simbolo “o”, quindi 65,74 diventa 65_o74.

Nel 1596 l'italiano Giovanni Antonio Mangini sostituisce il punto al simbolo “o”, infine nel 1608 l'olandese Willebrord Snell (traduttore di Stevin in latino) introduce la virgola al posto del punto.

Nel 16° secolo a causa delle esplorazioni geografiche e dell'emergente interesse per i moti della terra e dei pianeti (rivoluzione copernicana), il **sistema di numerazione indo-arabo** subisce una spinta definitiva e si afferma come **unico sistema** di numerazione.

Tycho Brahe nel tentativo di confutare la tesi di Copernico e dimostrare che tutti i pianeti orbitano intorno alla terra e la terra intorno al sole compie dei calcoli nei quali utilizza le **formule di prostaferesi** (formule inventate da Johann Werner agli inizi del 16° secolo per calcolare il prodotto di due funzioni trigonometriche mediante la somma altre due).

Tali calcoli vengono poi utilizzati da Keplero per formulare le sue leggi sulle orbite ellittiche dei moti dei pianeti attorno al sole.

Oltre alle formule di prostaferesi vengono sviluppate tante altre tecniche per **velocizzare e semplificare i calcoli**.

Un calcolo famoso che viene fatto in questo periodo è il calcolo dell'orbita di Cerere da parte di Gauss che l'ha fatto inventando la formula dei minimi quadrati.

Invece la scoperta di Nettuno è stata fatta grazie alle **equazioni differenziali**, grazie alle quali si è potuto osservare che il comportamento degli altri pianeti era influenzato dalla presenza di un corpo non ancora identificato.

Nello stesso periodo la navigazione ha una grossa spinta, si esplorano nuovi territori (si veda Cristoforo Colombo) ed è necessario fare velocemente i conti per disegnare le carte mentre si esplora.

Le tecniche di semplificazione dei calcoli però sono limitate e spesso non garantiscono

Etimologia di “tavola”:

L'utilizzo di questa parola per indicare una tabella è probabilmente dovuto ad un errore compiuto da un copista nel trascrivere l'Ars Geometrica di Boezio: esso disegnò un “quadro” di moltiplicazione al posto di una Mensa Pythagorica, un abaco di aspetto molto simile, ma lasciò la dicitura Tabula Pythagorica.

Errore ed etimologia sono stati rilevati intorno alla metà dell'ottocento.

un'efficienza sufficiente, per questo motivo vengono ampiamente utilizzate anche le **tavole numeriche** per fare i calcoli, ovvero tavole già precompilate con i risultati di molte operazioni.

Le tavole numeriche sono sempre state un ottimo strumento, ma spesso hanno **poche cifre significative o errori di calcolo**.

Le prime tavole vengono costruite per i logaritmi in base 10 da Henry Briggs (1561-1630), contenevano 1 200 000 cifre ed avevano al loro interno 603 errori (questo le rende un buono strumento, ma non 100% affidabile).

Calcolo filosofico:

Lullo (1233-1315) cerca di trovare quali sono i principi comuni a tutte le scienze (già Aristotele aveva distinto i principi di ogni scienza particolare da quelli comuni, ma non aveva indicato quali fossero), ovvero quei principi su cui si basa ogni argomentazione in un qualunque campo.

Trovati questi principi di base (che definisce nell'Ars Generalis) poi è possibile derivare tutte le proposizioni vere come calcolo combinatorio di questi termini.

Nell'Ars Combinatoria formula in maniera sistematica il metodo per la divisione di un problema in sottoproblemi via via sempre più semplici, questi problemi vengono poi etichettati con lettere dell'alfabeto ed in seguito manipolati calcolando.

Hobbes (1588-1679) ed i suoi seguaci tentano di sviluppare la teoria di Lullo sviluppando in particolare la parte relativa al ragionamento come calcolo.

Teorizza infatti che pensare è calcolare ed indica il sillogismo come una sorta di somma:

- Il giudizio affermativo è l'unione, o somma di due nomi: con esso si intende che i due termini (il soggetto e il predicato) sono nomi della stessa cosa; se si dice "l'uomo è ragionevole" significa che il termine "uomo" e il termine "ragionevole" sono due nomi dello stesso individuo.
- Il giudizio negativo è la separazione, o sottrazione di due nomi: con esso si intende che i due termini sono nomi di cose diverse.
- Il sillogismo, o ragionamento, è l'addizione di tre nomi: consiste nel constatare che il termine maggiore, il termine minore e il termine medio sono tre nomi diversi della stessa cosa.

Rimane però il problema dei nomi: se non si è concordi sul significato dei nomi non è possibile dare un senso a queste operazioni.

Pensa quindi che bisognerebbe produrre un vocabolario certificato dai filosofi dove ogni parola ha un significato preciso e senza ambiguità.

I filosofi dell'abbazia di Port Royal ragionano anch'essi sul linguaggio naturale sostenendo che ogni lingua è una semplice espressione del pensiero umano sottostante (quindi secondo loro il linguaggio utilizzato non influisce sul modo di pensare), di conseguenza cercano di identificare quelle regole comuni del pensare e del parlare umano. Facendo ciò cercano di ridurre i principi etici e filosofici ad un insieme di semplici norme e cercano di elaborare criteri razionali che guidino la conoscenza ed il giudizio critico.

Leibniz (1646-1716) invece rigetta il concetto di disciplinare il linguaggio naturale utilizzando proprio un intero **nuovo linguaggio formale** completamente diverso da quello naturale.

Per ideare ciò attinge ad un enorme bagaglio culturale che gli permette di conoscere molto bene i pregi e le limitazioni dei linguaggi naturali e formali, infatti lui stesso è storico, diplomatico, giurista, tecnico (ha progettato macchine) e matematico.

Inoltre trae forte ispirazione dall'utilizzo dei simboli nel linguaggio formale dell'I Ching, dalla praticità del sistema di numerazione indo-araba rispetto al romano e dalla sua efficienza nella manipolazione delle espressioni.

Presi questi spunti propone quindi di smettere di tentare di disciplinare il linguaggio naturale, ma di creare un nuovo sistema di scrittura con simboli digitali.

Questo sistema è un dispositivo linguistico formale per descrivere ogni tipo di argomentazione come manipolazione di simboli digitali, in maniera analoga a come l'aritmetica manipola cifre e simboli.

Leibniz espone le sue tesi in due opere principali:

- *Characteristica universalis*: determinazione sistematica e rigorosa dell'insieme dei concetti fondamentali rappresentati da simboli;
- *Calculus ratiocinator*: formalizzazione delle leggi sintattiche di combinazione dei segni tramite l'utilizzo di opportuni operatori.

e la sua filosofia su come dovrebbe essere questo linguaggio è espressa dalla celebre frase:

“Quo facto, quando orientur controversiae, non magis disputatione opus erit inter duos philosophos, quam inter duos computistas. Sufficiet enim calamos in manus sumere sedereque ad abacos, et sibi mutuo (accito si placet amico) dicere: **calculemus!**”

Ovvero (traduzione mia non necessariamente 100% corretta):

“Fatto ciò [ovvero definito il linguaggio formale], quando sorgerà una controversia, non ci saranno discussioni tra due filosofi più grandi di quelle tra due persone che fanno dei calcoli. Sarà infatti sufficiente prendere in mano la penna e sedersi all'abaco, ed entrambi dirsi a vicenda (chiamando se vogliono un amico [come testimone??]): calcoliamo!”.

La nascita delle macchine:

Per millenni l'unico dispositivo operativo è stato l'**uomo**, il quale però si è dotato di vari strumenti per semplificare il proprio lavoro (periferiche operative), come per esempio le tavolette di argilla, l'abaco, le tavole numeriche, i regoli ed il compasso.

Successivamente compaiono i primi dispositivi (pseudo)operativi: macchine meccaniche per fare le somme e sottrazioni (Pascal ne ha costruite alcune, ma anche altri), questi strumenti non sono ancora autonomi ma sono un primo passo in quella direzione.

Poco dopo viene costruita una macchina per le moltiplicazioni progettata da Leibniz.

Questi tipi di dispositivi sono formali speciali, infatti sono in grado di eseguire una sola operazione.

Con l'avvento delle **macchine a vapore** si ha un grande sviluppo dei dispositivi operativi che sfruttano l'energia meccanica.

Tralasciando alcuni tentativi senza seguito come l'eolipila di Erone o l'architruono di Leonardo da Vinci, il grosso sviluppo delle tecniche per sfruttare il vapore si ha tra il 1600 ed il 1700.

Nel 1606 gli esperimenti di Giovanni Battista della Porta riescono ad utilizzare il vapore come forza motrice.

Nel 1615 Salomon de Caus pubblica un trattato su un sistema basato su una pompa a vapore.

Nel 1679 Denis Papin inventa la pentola a pressione.

Nel 1705 con la macchina di Newcomen si ha il primo esempio di applicazione industriale, dove il vapore viene utilizzato per azionare macchinari per il trasporto dei materiali o per il pompaggio dell'acqua.

Nel 1765 con l'invenzione di **Watt** del **meccanismo biella-manovella** il vapore inizia ad essere utilizzato in larga scala a **livello industriale**, in quanto si riducono costi, dimensioni e consumi, mentre aumenta la potenza disponibile.

Questa valvola inventata da Watt si autocontrolla facendo uscire il vapore senza bisogno dell'intervento umano, questo è molto importante perché introduce il concetto di sistema in grado di autoregolarsi.

Il **telaio meccanico** (1787) di **Jacquard** è stato il primo strumento reso meccanico di grande diffusione e ciò con cui nasce la rivoluzione industriale.

Nel 1814 viene progettata da Stephenson la prima locomotiva.

Il primo telaio elettrico verrà poi costruito da Bonelli, Bolmida e Vicenza.

La nascita dell'informatica

La rivoluzione industriale nel calcolo:

Le rivoluzioni culturali hanno fatto aumentare enormemente l'esigenza di eseguire molti calcoli in tempi ragionevoli.

Questo però si poteva fare solo utilizzando le **tavole**, di conseguenza è necessario produrre tavole sempre più grandi, con più cifre significative e con meno errori.

Henry Briggs (1561-1630) pubblica le tavole dei logaritmi in base 10 dei naturali fino a 20 000 e da 90 000 a 100 000 a quattordici cifre decimali, l'intervallo mancante viene poi completato con 10 cifre decimali dall'olandese Adrian Vlacq.

In seguito su queste tavole verranno trovati 603 errori su oltre 1 200 000 cifre.

Edward Sang pubblica insieme alle figlie nel 1871 le tavole dei logaritmi da 1 a 10 000 con 28 cifre significative e da 100 000 a 200 000 con 15 cifre significative.

Nel 1814 compaiono tavole con la scomposizione in fattori primi, il quadrato, il cubo, la radice quadrata, il reciproco ed il logaritmo naturale dei numeri naturali da 1 a 10 000.

Nel mentre si diffondono ad uso dell'ingegneria tavole con meno cifre decimali.

Nel 1793 l'assemblea nazionale francese istituisce il sistema metrico decimale ed affida a **De Prony** il compito di realizzare tavole numeriche delle funzioni logaritmiche e trigonometriche con un numero di cifre variabile da 14 a 29 posizioni.

Questa impresa è la più ambiziosa che sia mai stata fatta in questo campo e richiede di adottare un approccio completamente diverso.

Con la rivoluzione industriale si sono diffuse le idee di Adam Smith sulla divisione del lavoro (di cui tratta nell'Indagine sulla natura e le cause della ricchezza delle nazioni, pubblicato nel 1776): la produttività aumenta notevolmente dividendo le mansioni in altre più semplici da assegnare a personale meno specializzato (se le mansioni sono sufficientemente semplici successivamente verranno assegnate alle macchine).

De Prony applica questi concetti della divisione del lavoro alla matematica: per produrre delle tavole divide le varie mansioni tra **matematici**, **tecnici** e **computisti** (in inglese computer).

I matematici fanno gli sviluppi in serie delle varie funzioni, da essi i tecnici producono dei polinomi ed i computisti effettuano le "tabulazioni" con le quali si calcola effettivamente il risultato.

Per fare queste tabulazioni è sufficiente saper fare delle somme, quindi i computisti sono personale molto poco specializzato.

I computisti (o computer) si possono quindi considerare dispositivi operativi non formali speciali (fanno solo somme e sottrazioni).

Per ogni polinomio y ($y = 1 + 2x + 3x^2 - x^3$) viene calcolata la parte iniziale delle differenze. Per polinomi di grado n , la differenza n -esima è costante, quindi si calcolano solo le prime $n+1$ righe. Nell'esempio $n = 3$.

x	y	y'	y''	y'''
0	1			
1	5	4		
2	9	4	0	
3	7	-2	-6	-6
4				-6

Sviluppo dei dispositivi linguistici:

Boole un centinaio di anni dopo Leibniz propone un **linguaggio formale per la manipolazione di simboli**.

Questi simboli rappresentano delle classi (la classe degli uomini, la classe dei viventi, la classe delle cose rosse, ecc...) e le regole per manipolarli traggono ispirazione dall'algebra.

Le operazioni che si possono fare tra le varie classi sono:

- +: unione;
- -: sottrazione;
- x: intersezione.

Esempio:

Se X rappresenta la classe delle cose bianche, Y quella degli animali e Z quella dei bipedi, allora $X \times Y$ è la classe degli animali bianchi e $X \times Y \times Z$ è la classe degli animali bianchi bipedi.

Inoltre Boole stabilisce che nella sua algebra 0 rappresenta la classe vuota, mentre 1 è la classe della totalità delle cose.

Utilizzando solo quelle due costanti si ottengono quindi le seguenti regole:

$$0 + 0 = 0 \quad 0 \times 0 = 0 \quad \underline{X + X = X} \quad \underline{X \times X = X}$$

$$0 + 1 = 1 \quad 0 \times 1 = 0 \quad \underline{X + 1 = 1} \quad X \times 1 = X$$

$$1 + 0 = 1 \quad 1 \times 0 = 0 \quad \underline{1 + X = 1} \quad 1 \times X = X$$

$$\underline{1 + 1 = 1} \quad 1 \times 1 = 1 \quad \underline{X + X = X} \quad \underline{X \times X = X}$$

Boole realizza poi che il suo dispositivo linguistico può essere utilizzato sia per il calcolo insiemistico che per il calcolo proposizionale cambiando unicamente il significato di alcuni simboli:

calcolo insiemistico	calcolo proposizionale
0 è l'insieme vuoto, 1 è l'insieme universo	0 è falso, 1 è vero
+ è l'unione, x è l'intersezione	+ è OR, x è AND

Quindi col linguaggio di Boole è possibile **calcolare il valore di verità di un'espressione** (si inizia a realizzare il "Quo facto [...] calculemus" di Leibniz).

Esempio:

Problema:

$$P = (A \sqcap B) \sqcap (C \sqcap D \sqcap E) \sqcap (A \sqcap D) \sqcap (B \sqcap E) = ?$$

Traduzione nel linguaggio di Boole:

$$(A + B) \times (C + D \times E) + (A \times D) + (B \times E) = ?$$

Introduzione dei dati: $A = C = 0$ e $B = D = E = 1$

$$(1 + 0) \times (0 + 1 \times 1) + (0 \times 1) + (1 \times 1) = 1$$

Si ottiene che $P=1$, quindi l'espressione è vera.

Inoltre con questo linguaggio è possibile calcolare i sillogismi, la logica quindi si inizia a trasformare da ragionamento (procedimento della mente umana) a calcolo (processo automatico).

Il linguaggio di Boole è un dispositivo linguistico formale generale, infatti con esso non è possibile esprimere tutti i tipi di espressione in quanto non ha i quantificatori (universale ed esistenziale), non prevede la presenza di variabili (non si possono fare equazioni, ovvero

avere un'incognita in una proposizione e manipolarla fino ad ottenerne il valore di verità) e non permette di esprimere funzioni il cui risultato dipende dalla proposizione data in input.

Frege espande l'algebra di Boole dandole più ampia capacità di espressione.

Il suo linguaggio chiamato Begriffsschrift (**ideografia** in italiano, ovvero modo di rappresentare graficamente le idee) ha:

- possibilità di descrivere proprietà delle proposizioni;
- regole per costruire proposizioni complesse (che prevedono l'utilizzo dei termini "tutti" o "alcuni");
- regole inferenziali per dedurre proposizioni vere partendo da altre proposizioni vere.

Nella logica di Frege esistono predicati e variabili: si possono avere enunciati del tipo $f(X)$ il cui valore 0 o 1 dipende dall'argomento X .

Quindi indicando con f una data proprietà, $f(X)$ è vera per ogni elemento X che ha quella proprietà.

Esempio:

Se h indica essere uomo, allora $h(\text{Socrate})$ è vera.

Inoltre con questo sistema si possono specificare le relazione tra due o più variabili:

indicando con g una relazione, $g(X,Y)$ è vera per ogni coppia (X,Y) per cui è vera g .

Sono inseriti poi anche i quantificatori universale (\forall) ed esistenziale (\exists).

Esempio:

Tutti i cavalli sono mammiferi

$(\forall x)(\text{cavallo}(x) \Rightarrow \text{mammifero}(x))$

Infine introduce la regola inferenziale del modus ponens: se una proposizione ne implica una seconda e questa prima è valida, allora è valida anche la seconda (se A allora B e A è vero, allora B è vero).

Esempio:

Tutti quelli che amano qualcuno sono felici

$(\forall x)((\exists y) \text{ Ama}(x,y) \Rightarrow \text{Felice}(x))$

Dati: Ama(William, Susy)

Esecuzione:

$x = \text{William}, y = \text{Susy}$

Deduzione \Rightarrow Felice(William)

Questo rende il linguaggio simbolico della **logica** un **dispositivo linguistico formale universale** (con Boole era generale).

Con questo linguaggio si può dimostrare che una formula è deducibile e trovare procedimenti per calcolare soluzioni.

Dal sistema formale di Frege verrà creato il linguaggio di programmazione Prolog da Robinson nella seconda metà del 1900.

Dal sistema di Frege nasce la convinzione che sia possibile, dato un sistema ben formato, dedurre da esso ogni possibile teorema e decidere se ogni proposizione è vera o falsa, questa teoria è detta **logicismo** e molti matematici vi lavorano per cercare di dimostrarla.

Hilbert (a cavallo tra il 1800 ed il 1900) formalizza ulteriormente la geometria fornendone una descrizione formale a partire da degli assiomi.

Nel mentre Russell e Whitehead fanno la stessa cosa con l'aritmetica cercando di derivarla a partire dalla logica, in questo modo danno origine alla logica di Frege-Russell.

Per dimostrare la fondatezza del logicismo, Hilbert propone due problemi:

- dimostrare che il sistema assiomatico per la logica di Frege-Russell è coerente e completo: ovvero che dai loro assiomi e con le loro regole inferenziali si possono derivare tutte (**completezza**) e sole (**coerenza**) le verità logiche.
- Trovare un metodo che, data una formula del sistema formale, sia in grado di decidere in un numero finito di passi se essa è vera o falsa, in altri termini dimostrare che il **problema della decisione** (Entscheidungsproblem) è risolubile con un algoritmo (**consistenza**).

Gödel dimostra che il sistema di Frege è completo e coerente (1° punto), però dimostra anche che il **problema della decisione non è risolubile** con un algoritmo e che in qualunque sistema formale sufficientemente espressivo si può sempre costruire una proposizione per la quale non può essere dimostrato in un insieme finito di passi se è vera o falsa.

Questo pone **fine** al **logicismo** in quanto mostra che un sistema non può essere completo, coerente e consistente contemporaneamente, ma è sempre necessario rinunciare ad una delle tre proprietà.

Confutata questa teoria ci si chiede quali siano le funzioni che possono effettivamente essere calcolate mediante un procedimento effettivo (**problema della calcolabilità**).

Gli obiettivi principali di questa ricerca sono:

- dare una definizione formale e matematicamente rigorosa dell'idea intuitiva di funzione calcolabile;
- individuare le categorie di problemi che sono effettivamente risolvibili (calcolabili).

Church individua un metodo per calcolare tutte le funzioni tramite la ricorsione (funzioni ricorsive primitive).

Anche Post e Markov sviluppano dei sistemi per individuare funzioni calcolabili.

Turing inventa un suo metodo per calcolare alcuni tipi di funzioni (**la macchina di Turing** e le **funzioni Turing-complete**) che è molto importante per la sua capacità di essere elementare (manipola simboli in maniera molto semplice) e di conseguenza si distacca dai sistemi precedenti che erano enormemente complessi.

Nel 1936 pubblica l'articolo "On Computable Numbers with an Application to the Entscheidungsproblem" dove espone il suo metodo.

Viene poi verificato che il metodo di Church e quello di Turing individuano lo stesso insieme di funzioni.

Nel sistema di Turing per eseguire una particolare manipolazione di simboli digitali si deve:

- Riportare su nastro (monodimensionale) i caratteri che identificano i dati iniziali;
- Predisporre l'elenco dei singoli stati che definiscono il tipo di manipolazione sui simboli riportati sul nastro;
- Indicare lo stato iniziale;
- Indicare la posizione del primo carattere da trattare.

Gli stati definiscono tre tipi di azioni:

- azione sul carattere corrente, per eventuale sostituzione con altro carattere;
- eventuale spostamento di un posto a destra o a sinistra;
- scelta dello stato successivo (che può anche rimanere lo stesso).

Quindi un istante i -esimo della computazione può essere rappresentato dalla quintupla:

- S_i : sigla che identifica lo stato corrente;
- X_i : il simbolo letto sul nastro;
- Y_i : il simbolo da sostituire a X_i ;
- V_i : spostamento (destra/sinistra) per scegliere il prossimo carattere;

- S_{i+1} : sigla che identifica lo stato successivo.

Ogni procedimento di soluzione descritto con metodo di Turing si chiama macchina di Turing.

Turing propone una macchina particolare che descrive come eseguire una qualsiasi macchina di Turing descritta sul nastro, questa macchina è detta **macchina universale di Turing** (MUT).

La macchina universale di Turing è un dispositivo linguistico formale universale.

Macchine calcolatrici:

Müller, ispirato dal telaio di Jacquard, è il primo ad ideare una macchina alle differenze per risolvere equazioni, ma la sua rimane un'idea e non viene mai effettivamente sviluppato un progetto.

Poco dopo **Babbage** progetta una **macchina per le differenze** (NON macchina differenziale, questo nome è sbagliato) la cui costruzione però non viene mai terminata in quanto, a causa della tecnologia dell'epoca non ancora sufficientemente sviluppata, ci sono continui intoppi ed alla fine terminano i finanziamenti del governo britannico.

Infatti Babbage era riuscito a farsi finanziare dal governo britannico proprio perché lo scopo della sua macchina era quello di creare **tavole numeriche in maniera automatica** (argomento su cui c'era molto interesse perché permetteva di ridurre notevolmente la quantità di errori).

Due esemplari di questa macchina verranno realizzati soltanto molti anni dopo in Svezia nel 1853 da Georg Scheutz e suo figlio, uno di essi verrà acquistato da un Ufficio Navale degli Stati Uniti ma si mostrerà un totale fallimento a causa della bassa efficienza.

Questa macchina è in grado di eseguire un solo tipo di operazione, quindi è un dispositivo operativo formale speciale, paragonabile ad un moderno elettrodomestico non programmabile.

Con le proprie risorse Babbage progetta poi la **macchina analitica**, ovvero una macchina in grado di fare tutte e quattro le operazioni dell'aritmetica, la quale però rimane solo un progetto teorico.

Successivamente Ada Lovelace si interessa al progetto e ne finanzia una parte.

Nella macchina analitica:

- La memoria per i dati è costituita da pile di dischi sulla cui superficie laterale sono riportate le cifre del sistema decimale, per inserire l'input prima dell'operazione si muovono manualmente le pile di input in modo che rappresentino il numero desiderato.
- L'**operazione** da eseguire è descritta su **schede perforate** (ispirate alle schede perforate del telaio meccanico).
- Per quanto riguarda il controllo delle istruzioni la sequenza è determinata dall'ordine delle schede perforate.
- La (pseudo) CPU, detta mulino, esegue l'operazione richiesta dalla scheda istruzione utilizzando le pile contenenti i dati richiesti e memorizza il risultato sulla pila di output, come specificato nella scheda istruzione stessa.

La macchina analitica è quindi un dispositivo operativo formale generale.

Nello stesso periodo (1831) a Torino Plana costruisce il calendario universale, ovvero una macchina analogica in grado di calcolare la collocazione settimanale di ogni giorno dell'anno con anche dati relativi a fasi lunari, maree, festività e santi del giorno, dall'anno 0 fino al 4000.

Questa macchina ha un funzionamento interno molto simile ai progetti di Babbage ed è indice del forte interesse che c'è a Torino per le macchine automatiche.

Per questo motivo Babbage viene invitato a Torino presso l'Accademia delle scienze dove illustra il progetto della sua macchina.

Menabrea assiste alla conferenza e da essa sviluppa un testo in cui definisce in maniera formale per la prima volta il linguaggio macchina di quella macchina (quindi non descrive come è fatta, ma come usarla).

Babbage vede il lavoro di menabrea e ne fa tradurre l'articolo in inglese da Ada Lovelace, la quale oltre a tradurlo lo espande con nuovi programmi come esempi.

Questo articolo ha un grande successo.

Il linguaggio di Menabrea ed Ada Lovelace è un dispositivo linguistico non formale generale.

Menabrea ed Ada **Lovelace** quindi sono i **primi programmatori** della storia dell'informatica.

Negli anni successivi l'utilizzo dell'elettricità da parte dell'uomo ha uno sviluppo enorme:

Volta costruisce la prima pila e tantissime cose diventano elettriche.

Nel 1890 viene creata la prima macchina elettrica in grado di fare operazioni più velocemente degli esseri umani.

L'effettivo utilizzo e sviluppo di questo tipo di macchine avviene negli uffici, prima con macchine meccaniche per fare operazioni matematiche ed in un secondo momento con macchine elettromeccaniche con lo stesso scopo.

Inoltre si diffondono dispositivi che sfruttano l'elettricità come il telegrafo e successivamente il telefono.

Hollerith costruisce (tra il 1884 ed il 1889) una **macchina selezionatrice** per manipolare delle schede, quindi in questo caso non si ha il programma scritto su schede ma i dati.

Attorno al 1890 l'ufficio che gestisce i censimenti degli stati uniti adotta una macchina di Hollerith grazie alla quale si riescono ad elaborare tutti i dati in meno di un anno (per elaborare i dati del 1880 c'erano voluti 8 anni).

Questo risultato viene notato dagli altri stati e dai privati che iniziano a voler adottare macchine dello stesso tipo.

Negli uffici si diffondono quindi una serie di macchine specializzate per la gestione dei dati su schede perforate come selezionatrici, perforatrici, riproduttrici (duplicano le schede), stampanti, ecc...

Queste macchine hanno un successo ancora più grande quando Hollerith fa una modifica: inserisce un pannello per modificare il tipo di selezione che la macchina poteva fare (è solo una macchina selezionatrice), quindi la macchina diventa programmabile (generale) ed ha diffusione su scala mondiale.

Un'ulteriore modifica viene fatta successivamente permettendo alla macchina di fare le operazioni elementari e di perforare sulla stessa scheda dove c'erano le istruzioni il risultato.

Hollerith grazie all'esperienza accumulata fonda nel 1896 la Tabulating Machine Company, che successivamente (1924), in seguito alla fusione con altre aziende, viene rinominata in **International Business Machines Corporation (IBM)**.

Comrie nel 1929 realizza che queste macchine possono essere utilizzate in **ambito scientifico** per elaborare grandi quantità di dati, di conseguenza le utilizza per calcolare la traiettoria delle luna in un arco molto ampio di tempo e per fare ciò utilizza 500 mila schede perforate.

Questo stimolo viene raccolto da Wallace J. **Eckert** presso la Columbia University che inizia a sviluppare insieme alla IBM una serie di macchine meccanografiche per il calcolo scientifico che diventeranno in seguito il sistema **Mark 1** e la **serie 600** della IBM.

Le macchine di Eckert sono in grado di eseguire i cicli for.

Konrad **Zuse** nel 1934 in Germania inizia ad occuparsi di un progetto per la realizzazione di un calcolatore automatico (uno dei pochi progetti in Germania), delle sue macchine è rimasto poco perchè sono state distrutte nei bombardamenti.

Zuse nel 1936 costruisce lo Z1, una macchina governata da un vero e proprio programma (non operazioni predefinite dell'hardware) scritto su un nastro di carta perforata.

Era in grado di elaborare numeri in virgola mobile ed aveva 16 registri di memoria realizzati con dei relè (di seconda mano riciclati dalle imprese di telefonia).

Lo **Z3 del 1941** è la **prima macchina turing completa** mai realizzata.

Questa macchina ha:

- Rappresentazione binaria dei numeri con virgola mobile;
- Programma memorizzato su nastro come sequenza di operazioni ciascuna rappresentata da codice operativo, due indirizzi degli operandi e un indirizzo per il risultato;
- Una memoria meccanica per i dati costituita da 64 registri con indirizzo numerico;
- Un dispositivo meccanico con tastiera per l'input collegato via cavo alla memoria, ogni dato registrato sulla tastiera viene trasmesso e memorizzato in uno dei registri della memoria.

Inoltre lo Z3 è programmabile in un **linguaggio di alto livello** (scritto sulle schede perforate) chiamato **Plankalkül**.

Usando questo linguaggio viene scritto un programma per il gioco degli scacchi.

Purtroppo il lavoro di Zuse non viene particolarmente conosciuto all'estero e la sua ricerca non influenza in maniera significativa lo sviluppo delle macchine a livello mondiale (in particolare negli Stati Uniti).

Tornando negli Stati Uniti il primo progetto di una **macchina calcolatrice elettronica** è quello dell'**ABC** del 1942 (Atanasoff Berry Computer), realizzato da **Atanasoff** e **Berry**.

È una macchina per eseguire semplici conteggi su degli eventi nucleari (quindi non è general purpose), di conseguenza a causa della necessità di eseguire i calcoli in maniera rapidissima per seguire tali eventi viene utilizzata per la prima volta l'elettronica.

Gli eventi bellici però separano i due ricercatori ed i loro progetti rimangono incompiuti.

Mauchly riprenderà questi progetti successivamente per realizzare l'ENIAC.

Un'altra macchina calcolatrice elettronica degna di nota è **Colossus**, un calcolatore elettronico digitale (non general purpose) costruito tra il 1943 ed il 1945 dall'Inghilterra per decifrare i messaggi tedeschi criptati con Enigma.

I suoi progettisti sono l'ingegnere Tommy Flowers ed il matematico Max Newman, che hanno utilizzato per costruirla anche dei lavori di crittoanalisi di Alan Turing (il quale ha lavorato alla Bomba Elettromeccanica, una macchina polacca con lo stesso scopo di Colossus, ma non ha mai contribuito direttamente a questa).

Questa macchina ed i suoi modelli successivi sono stati distrutti alla fine della guerra e la loro esistenza è stata tenuta nascosta fino a metà degli anni 70, di conseguenza ha contribuito poco allo sviluppo mondiale delle macchine (come i calcolatori di Zuse).

Aiken nel 1943 convince l'IBM a collaborare per realizzare l'ASCC (Automatic Sequence Controlled Calculator) che è una macchina di Babbage elettronica, con lo scopo di utilizzarla nei calcoli scientifici per risolvere equazioni differenziali e dimostrare che il progetto di Babbage era consistente.

Questa macchina viene poi chiamata **Mark 1** e rimane attiva ad Harvard dal 1944 al 1959. Con lo scoppio della guerra viene utilizzata dal governo americano per calcolare le traiettorie dei proiettili.

Il design di questa macchina però è già vecchio: il programma memorizzato su un nastro di carta non prevede istruzioni di salto di alcun tipo, rendendo impossibile avere costrutti condizionali o cicli; di conseguenza Aiken inizia a lavorare a versioni successive (Mark 2, 3 e 4) in cui introduce tra le altre cose anche il salto condizionato.

Aiken però non riesce a vedere altri possibili utilizzi delle macchine elettroniche oltre a calcolare traiettorie di proiettili e fare calcoli scientifici, di conseguenza non riesce a convincere l'IBM a proseguire nel progetto e deve proseguire da solo, infatti l'IBM è più interessata al mercato aziendale con macchine meccanografiche a schede come quella di Hollerith.

Negli stessi anni la **Bell** (azienda di telecomunicazioni) inizia a produrre esemplari di macchine che realizzano collegamenti tra un computer ed un **terminale remoto** telescrivente.

La prima realizzazione significativa di calcolo a distanza è rappresentata dai totalizzatori usati nelle corse ippiche.

Le scommesse venivano inizialmente raccolte in vari punti dell'ippodromo ed erano trasmesse ad una centrale dove venivano accumulate: per ogni cavallo esisteva un recipiente e il messaggio in arrivo faceva scattare un meccanismo che faceva cadere una biglia d'acciaio nel recipiente appropriato.

Nel 1930 sono comparsi totalizzatori elettrici trasportabili utilizzabili per 6 cavalli e capaci di registrare fino a 12000 scommesse al minuto (quindi niente più biglie, ma memorizzazione elettronica).

Nel 1937 presso i Bell Telephone Laboratories G. Stibitz inizia a costruire una macchina calcolatrice capace di moltiplicare e dividere numeri complessi.

Questa macchina, modificata con l'aggiunta dell'addizione e sottrazione e collegata fino a tre telescriventi, viene chiamata "Complex Number Computer" ed è rimasta in uso fino al 1949.

Con l'inizio della guerra, i laboratori Bell sono stati coinvolti nella costruzione di una serie di calcolatori a relè dotati di circuiti di autocontrollo chiamati Model I, Model II e Model III; quest'ultimo, controllato da un programma registrato su nastro, è stato completato nel 1944 ed è rimasto in funzione fino al 1958.

Il Model V è un calcolatore "general purpose" controllato da programma con istruzioni di controllo condizionato; può mantenere in esecuzione due programmi contemporaneamente gestendo in modo automatico il controllo fra i due.

Nel mentre l'IBM, abbandonata la linea del Mark 1, sperimenta la serie CPC (Card Programmed Calculator) che è più simile alle macchine da ufficio di Hollerith.

Contemporaneamente continuando a collaborare con Eckert nasce **SSEC** (Selective Sequence Electronic Calculator) che utilizzando **tubi a vuoto** (grande innovazione) rimane in attività dal 1948 al 1952.

Grazie alla serie 600, SSEC e CPC la IBM accumula sufficiente esperienza per dominare il mercato.

Il primo tentativo di utilizzo dell'elettronica per costruire un computer general purpose viene fatto da **Vannevar Bush** nel 1937 presso il MIT con il **Rapid Arithmetical Machine Project**. In questo progetto è previsto l'utilizzo dell'elettronica per l'aritmetica ed una piccola parte della memoria.

Altri progetti importanti per quanto riguarda i calcolatori elettronici sono stati fatti presso la Moore School of Electrical Engineering della University of Pennsylvania, impegnata a collaborare col Ballistic Research Laboratory per calcolare le tavole di tiro con l'utilizzo di un analizzatore differenziale (versione avanzata di quello costruito da Bush nel 1937).

In questi progetti è coinvolto John **Mauchly** che nel 1942, dopo aver studiato il lavoro di Atanasoff, scrive un **rapporto** in cui si dichiara a favore dei **calcolatori "a impulsi"** (digitali) rispetto a quelli analogici e descrive la possibilità di costruire un calcolatore elettronico digitale.

Questo rapporto dà il via al progetto **ENIAC** (Electronic Numerical Integrator and Computer) che nel 1943 viene approvato dal governo americano.

Col contributo di Eckert la costruzione della macchina termina nel 1945 e nel gennaio del 1946 viene messa in funzione presso la Moore School, poi viene trasferita al Ballistic Research Laboratory dove rimane in funzione fino al 1955.

L'ENIAC è il primo calcolatore general purpose interamente elettronico.

Compare però subito un problema: l'ENIAC non ha un linguaggio di programmazione ma il suo programma è specificato cambiando manualmente i **collegamenti hardware**, di conseguenza scrivere i programmi è estremamente complicato.

Questa difficoltà dà il via al progetto **EDVAC** prima ancora che il progetto ENIAC termini.

Nello stesso periodo Von Neumann viene a conoscenza di ENIAC e ne comprende il potenziale.

Computer	luogo	Operativa	Binario	Elettronico	Programmabile	Turing completo
Complex Number Calculator (Stibitz)	New York	1940 demo a distanza	Si	No	Parzialmente	No
Z3 (Zuse)	Berlino	Maggio 1941	Si	No	Tramite schede perforate	Si (1998)
ABC (Atanasoff - Berry) Computer	Iowa University	Estate 1941	Si	Si	No	No
Colossus (Flowers e Newman)	Bletchley	Dicembre 1943 /	Si	Si	Parzialmente, ricollegando	No

		Gennaio 1944			parti hardware	
Harvard Mark I/IBM ASCC (Aiken)	Harvard	1944	No	No	Tramite schede perforate	No
ENIAC (Eckert e Mauchly)	Moore School	1944	No	Si	Parzialmente, ricollegando parti hardware	Si
	Aberdeen	1948	No	Si	Tramite Function Table ROM	Si
EDVAC (Eckert, Mauchly e von Neumann)	Aberdeen	1949	Si	Si	Si	Si
Manchester Mark 1 (Williams e Kilburn)	Manchester	1949	Si	Si	Si	Si
EDSAC (Wilkes)	Cambridge	1949	Si	Si	Si	Si

La fusione delle discipline

Prima del 1945 lo sviluppo della **logica** (problema della decisione, problema della calcolabilità, sviluppo di linguaggi formali universali, ecc...) e lo sviluppo delle **macchine** hanno proceduto su binari paralleli come **discipline separate**.

Senza la contaminazione di questi due campi non possono nascere le macchine general purpose ed i linguaggi di programmazione universali.

Questa situazione cambia con le dimostrazioni di Shannon e con l'avvento della cibernetica.

Shannon:

Prima del 1935 l'elettronica non viene mai presa in considerazione per costruire macchine calcolatrici perchè le valvole elettroniche avevano tempi di vita talmente brevi che non sarebbero arrivate integre a fine calcolo (anche per operazioni elementari), inoltre i calcoli fatti con i circuiti elettronici avevano delle probabilità non indifferenti di restituire un risultato errato.

Tra il 1935 ed il 1945 circa il metodo con cui vengono fatte le valvole migliora e la loro durabilità aumenta.

Shannon negli stessi anni fa una serie di pubblicazioni introducendo 4 concetti che cambieranno per sempre il mondo dell'elettronica e dell'informatica.

Nel 1938 pubblica "A Symbolic Analysis of Relay and Switching Circuits" (un'analisi simbolica dei relè e dei circuiti), una tesi di laurea in cui dimostra che un segnale elettrico in una rete di interruttori segue esattamente le regole dell'algebra di Boole, facendo corrispondere il valore "vero" allo stato "aperto" ed il valore "falso" allo stato "chiuso".

Pertanto un **circuito digitale** può essere descritto da un'**espressione booleana**, la quale può essere manipolata senza cambiare significato secondo le regole della sua algebra.

Da qui dimostra che si può fare la somma di due numeri binari in quanto ciò è descrivibile con l'algebra di Boole (per provarlo costruisce un addizionatore elettronico).

Nel 1948 pubblica il saggio "Una teoria matematica della comunicazione" dove conia la parola "**bit**" e rende canonica la **rappresentazione binaria** dell'informazione (è uno dei primi ricercatori della teoria dell'informazione).

Inoltre teorizza un utilizzo della **ridondanza** per risolvere il problema del **rumore** (scarsa attendibilità) dei circuiti elettrici: se un circuito ha una probabilità di restituire un risultato errato inferiore al 50% nell'unità di tempo, è sufficiente ripetere più volte il calcolo (un numero di volte N che si può ottenere con dei calcoli) e prendere il risultato che si ottiene più volte.

Questo modo di utilizzare i circuiti fornisce più garanzie di correttezza ed è comunque molto più veloce delle macchine analogiche.

Infine formula il **teorema del campionamento**: è possibile sostituire un segnale continuo (analogico) con una serie di campioni (digitali) su intervalli regolari sufficientemente brevi ed ottenere un risultato che ha un errore più piccolo di quello tollerato (anche le macchine analogiche hanno una certa precisione e di conseguenza un errore).

Quindi in altre parole se i campioni sono raccolti con una frequenza sufficientemente elevata questi forniscono tutta l'informazione contenuta nel segnale continuo.

Con questo dimostra che la **rappresentazione digitale delle informazioni può sostituire quella analogica**.

Cibernetica:

Nel decennio 1936-1946 nasce la cibernetica.

La **cibernetica** è una disciplina che si occupa dello studio dei processi di comunicazione e controllo dell'informazione negli animali e nelle macchine.

Questa branca della scienza si pone come obiettivo quello di costruire delle macchine che simulino la **capacità cognitiva** (comparare il concetto di macchine **general purpose**).

La cibernetica quindi si propone sia di costruire macchine per la simulazione dell'attività cerebrale, sia di utilizzare tali macchine per comprendere meglio queste attività di controllo e comunicazione.

Questa disciplina attinge dalla matematica, logica, fisica e neurologia, quindi ha la capacità di far **convergere campi** che fino a quel momento non si sono mai toccati tra loro.

Turing e Von Neumann nella loro carriera si occupano anche di cibernetica.

Nel 1906 Golgi riceve il premio Nobel per aver scoperto la struttura del neurone.

Nel 1943 **McCulloch e Pitts** presentano una descrizione astratta dei meccanismi regolatori del neurone, il quale è rappresentato da una struttura con input, output ed un'unità di elaborazione che se gli input raggiungono una certa soglia di intensità restituisce un output, altrimenti non fa nulla.

Questo è un primo esempio di **neurone artificiale**.

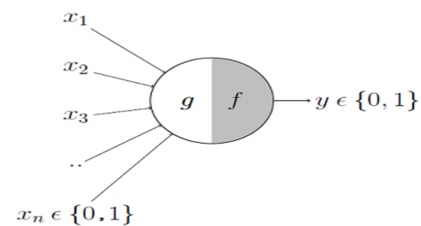
McCulloch e Pitts forniscono anche le implementazioni di tutte le porte logiche con questi neuroni.

I neuroni si possono collegare a catena tra di loro per avere comportamenti complessi (per esempio una rete di questi neuroni può descrivere l'algebra di Boole).

Da questa rappresentazione formale si inizia a lavorare per fare un modello fisico funzionante, da combinare ad altri modelli per far emergere i comportamenti più complessi tipici del cervello.

Quindi l'idea di intelligenza artificiale nasce molto presto nello sviluppo dei computer.

Von Neumann realizza che la valvola elettronica (o tubo a vuoto) è il corrispettivo fisico del modello di McCulloch.



La fusione di Von Neumann:

Von Neumann è un personaggio eclettico, che raccoglie in sé le esperienze di **molti campi di ricerca** e per questo è in grado di avere una visione di più ampio respiro sullo sviluppo dei calcolatori.

Infatti è un matematico che si occupa ampiamente di teoria dei giochi, ha lavorato su problemi di analisi numerica per soluzioni approssimate (conoscendo anche le loro

automatizzazioni da Babbage ad ENIAC), è impegnato nel progetto Manhattan come fisico (esigenza di fare calcoli impegnativi), è un esperto di cibernetica e nel campo della logica conosce molto bene il logicismo (con il suo fallimento) e le teorie sulla calcolabilità di Turing. Inoltre conosce le dimostrazioni di Shannon sull'elettronica.

Von Neumann viene coinvolto nella progettazione dell'**EDVAC** (Electronic Discrete Variable Arithmetic Computer), il successore di ENIAC.

L'articolo sul neurone artificiale di McCulloch e Pitts gli fa realizzare che EDVAC può essere pensato come qualcosa di molto **simile** ad un **sistema nervoso**:

- entrambi i sistemi elaborano informazioni;
- i triodi elettronici (tipi particolari di valvole) svolgono la stessa funzione dei singoli neuroni.

Da qui Von Neumann inizia a pensare al computer come un "cervello elettronico".

Partendo da questi presupposti definisce quindi per EDVAC una **architettura** (che poi diventerà il suo famoso modello) che ha:

- una sola memoria condivisa da istruzioni e dati,
- un solo bus per accedere alla memoria,
- una unità per l'aritmetica,
- una unità per gestire l'esecuzione delle istruzioni.

Questo permette alla macchina di **modificare** ed aggiornare parte del **programma a runtime** in quanto si trova in memoria, nonché di avere un linguaggio Turing completo.

Per quanto riguarda l'aspetto linguistico ispirandosi al modello della macchina di Turing **definisce in modo formale il linguaggio** della sua macchina, indicando che deve:

- avere le espressioni aritmetiche,
- avere tutte le primitive che gli consentono di essere Turing completo (if, while...),
- avere la possibilità di modificare parte delle istruzioni a runtime, quindi istruzioni che permettono di manipolare altre istruzioni.

In pratica le istruzioni, lunghe 20 bit, sono memorizzate a coppie in una parola di 40 bit.

Dei venti bit di ciascuna istruzione, otto sono usati per il codice operativo, mentre i restanti 12 ospitano un indirizzo di una parola della memoria (avere gli indirizzi nell'istruzione è utile in quanto il programma si può automodificare e cambiare quei valori a runtime).

Da questo momento in poi nascono i concetti di compilatore, assemblatore, interprete, loader e debugger.

EDVAC quindi è la **prima macchina elettronica general purpose turing completa**.

Il suo linguaggio macchina è un dispositivo linguistico formale universale, mentre lei è un dispositivo operativo formale universale.

il problema di EDVAC è che ha solo due registri nella cpu e di conseguenza si crea un collo di bottiglia e si perde molto tempo a trasferire dati o a modificare le istruzioni in memoria.

Questa cosa verrà corretta nelle macchine successive potenziando l'hardware ed adottando dei linguaggi macchina RISC (Reduced Instruction Set Computers) che riducono il numero dei riferimenti in memoria da parte delle operazioni.

Le specifiche relative al linguaggio (come sono costruite le istruzioni e cosa devono fare) vengono poi diffuse nel tentativo di creare uno standard al quale si possano adattare anche gli altri progettisti di macchine calcolatrici; questo standard è chiamato **IAS standard** (Institute for Advanced Study, l'ente presso il quale lavora Von Neumann).

Esso ha successo e vengono costruite in tutto il mondo 15 macchine con hardware diverso e linguaggi equivalenti (ma non 100% uguali).

In seguito alla costruzione dell'EDVAC, Vannevar Bush pubblica nel 1945 l'articolo "as we may think" in cui riassume il nuovo concetto di linguaggio di programmazione, in particolare afferma che:

- "A good programming language is a conceptual universe for thinking about programming".
- "A language that doesn't affect the way you think about programming, is not worth knowing".
- "There will always be things we wish to say in our programs that in all known languages can only be said poorly", ovvero si possono trovare linguaggi sempre più specifici.

Il computer entra nella società

Hardware:

Finita la seconda guerra mondiale esistono molte attività che utilizzano la meccanografia come supporto.

Per quanto riguarda invece i calcolatori elettronici (detti super computer) si inizia a dibattere su quale possa essere la loro utilità, infatti vengono considerati troppo potenti per quelle che sono le necessità di tutti i giorni e sono visti come “mostri” del calcolo.

L'opinione principale è che questi computer elettronici non avrebbero avuto un grande successo a livello industriale, ma che ne sarebbero serviti pochi a livello mondiale per fare i calcoli più difficili e per calcolare le tavole.

A riprova di ciò Aiken (all'epoca progettista alla Harvard Machine) afferma: “Non vedo come questi computer [che integrano equazioni differenziali] possano essere utili per gestire pratiche commerciali”.

Eckert e Mauchly dopo l'esperienza con ENIAC si mettono in proprio nella produzione dell'**UNIVAC** (universal automatic computer), che ha come scopo quello di creare il primo computer che non ha scopi militari ma civili.

Iniziano a lavorarci nell'aprile del 1946 finanziati dall'esercito americano, ma sottostimano i costi ed i tempi, smettono di ricevere finanziamenti e giungono vicini al fallimento.

A salvarli interviene l'azienda Remington Rand Inc. (marca di rasoi) e la costruzione termina nel 1951 con 46 esemplari (con più del doppio delle spese stimate ed in molto più tempo di quello stimato).

La particolarità di questa macchina è l'intuizione di darle uno **scopo universale** e non solo quello di calcolare tavole.

L'UNIVAC diviene famosa con le elezioni presidenziali degli Stati Uniti del 1952, nelle quali viene utilizzata per elaborare dati statistici e prevedere quale presidente avrebbe vinto.

Di conseguenza inizia ad insinuarsi nell'opinione pubblica l'idea che queste macchine possano avere un utilizzo anche in campo non militare.

Il ruolo dell'IBM:

L'IBM, evolutasi dalla società di Hollerith, è l'azienda dominante nel campo delle macchine meccanografiche.

Le sue grandi dimensioni le permettono di seguire contemporaneamente due linee di sviluppo:

- linea industriale per l'automazione degli uffici,
- linea sperimentale finanziando e realizzando grandi progetti innovativi.

Per quanto riguarda la **linea industriale** con la produzione della **serie 600** si ha un graduale passaggio da macchine meccanografiche a calcolatori elettronici.

L'IBM 601 del 1931 è a tutti gli effetti una macchina meccanografica in grado di eseguire moltiplicazioni e di perforare il risultato su schede.

L'IBM **603** del 1946 è il primo **calcolatore elettronico** prodotto in serie (ne vengono costruiti un centinaio di esemplari), legge ancora l'input da schede e restituisce il risultato sulle stesse, può eseguire addizioni, moltiplicazioni e divisioni (queste ultime mai usate perchè il risultato ha una precisione di solo tre cifre).

L'IBM 604 del 1948 è una versione migliorata della 603 con una CPU del tipo CPC, più modulare, con programmi di al massimo 60 istruzioni ed in grado di fare divisioni sufficientemente precise.

Queste caratteristiche le fanno avere molto successo e ne vengono costruiti oltre 5000 esemplari.

Con l'IBM 608 del 1955 le valvole vengono sostituite da transistor, rendendo la macchina meno ingombrante ed in grado di consumare meno energia.

L'IBM 610 è invece il primo computer costruito per uso personale negli uffici, è la prima macchina ad avere una tastiera come dispositivo di input (non più solo schede).

Von Neumann fa da consulente all'IBM per la costruzione della macchina **650**, ovvero la risposta dell'azienda all'UNIVAC.

La memoria centrale della 650 è realizzata con un tamburo elettromagnetico con 2000 parole indirizzabili, mentre la CPU ha un registro con l'istruzione da eseguire, un registro distributore utilizzato per spostare le parole da un posto all'altra ed un registro per il risultato delle operazioni grande il doppio (diviso in parte upper ed una lower).

Oltre al linguaggio macchina viene fornito un **linguaggio interpretato** di più alto livello, questo fa diffondere enormemente la figura del programmatore.

L'invasione della Corea del Sud da parte della Corea del Nord e della Cina (1950) fa tornare in stato di allarme gli Stati Uniti che investono di nuovo grosse somme di denaro nello sviluppo di computer, l'IBM sfrutta l'occasione per farsi finanziare lo sviluppo delle sue macchine, in particolare la **serie 700**.

L'IBM **701** (1951-1953) è una macchina costruita per scopi militari e viene venduta principalmente agli organi governativi americani (ne vengono costruiti solo 19 esemplari). Una di queste macchine sostituisce l'SSEC di Eckert.

Per la prima volta oltre alle schede perforate vengono utilizzati anche i nastri magnetici per memorizzare le informazioni.

È il primo computer disponibile sul mercato ad avere il **programma in memoria**.

Il suo successore è l'IBM 704 (1954) che ha hardware migliorato ed ha in dotazione un compilatore per il fortran.

Per questa macchina viene scritto il primo programma di "self-learning" che impara autonomamente a giocare a dama.

Le sue vendite sono molto maggiori, infatti ne vengono vendute 123 copie; una di esse viene installata nel 1961 a Bologna presso il CNEN (comitato nazionale energia nucleare).

Riassumendo si può quindi affermare che nell'evoluzione dell'hardware si sono susseguite cinque generazioni di macchine:

- **Generazione zero** (1940-1950): le **prime macchine**, EDVAC, EDSAC, IAS;
- **Prima generazione** (1951-1959): utilizzo di valvole, UNIVAC, IBM 650, 701, 704, questa generazione ha periferiche molto lente;
- **Seconda generazione** (1959-1965): utilizzo di **transistor**, memorie a nuclei di ferrite e dischi magnetici, viene ideata la memoria virtuale e le periferiche sono più veloci.

- **Terza generazione** (1965-1971): utilizzo di **circuiti integrati**, terminali, memorie esterne smontabili, viene costruita l'IBM 360 e compatibilità tra modelli, il Byte diventa standard;
- **Quarta generazione** (1971-1979): **microprocessori**, terminali (semi)autonomi, reti di calcolatori, supercomputer, mini computer ed i primi personal computer.

Software:

Dalla terza generazione in poi i linguaggi macchina iniziano a diventare troppo complessi e la memoria diventa più facilmente manipolabile, di conseguenza iniziano a svilupparsi in maniera sempre maggiore i **linguaggi di alto livello** ed una serie di **software** ad essi connessi.

Tra questi software si hanno:

- **Loader**: programma per caricare altri programmi e librerie in memoria.
- **Assemblatori e compilatori**: i primi evitano l'uso dei codici binari nella scrittura dei programmi applicativi, i secondi facilitano la scrittura e la formulazione dei programmi applicativi consentendo l'uso di linguaggi di alto livello.
- **Sistemi operativi**: programmi per facilitare l'input/output di insiemi di programmi e per ottimizzare l'utilizzo delle risorse.

I compiti dei sistemi operativi sono:

- Multiprogrammazione e multitasking: inizialmente con elaborazione batch, che permette a più programmi di eseguirsi contemporaneamente ma non garantisce l'interattività.
- Time sharing
- Real time
- Memoria virtuale
- Gestione della comunicazione (tra cui TCP/IP)

Negli stessi anni nasce il concetto di **database** e di DBMS che si evolve fino a delineare tre modelli diversi:

- gerarchico (molto usato fino alla fine degli anni '80),
- reticolare (mai veramente implementato),
- relazionale (ideato da Edgar Codd nel 1970 ed implementato in maniera sempre poco fedele alla definizione originale).

L'utente del computer:

Con la diffusione delle macchine a livello mondiale e con i linguaggi di alto livello nasce il **programmatore** come figura separata dal costruttore della macchina.

Anche gli utenti si possono dividere in cinque generazioni (simili ma non uguali a quelle dell'hardware):

- **Generazione zero** (1940-1950): gli utenti sono gli stessi **progettisti** e costruttori dell'hardware.

- **Prima generazione** (1951-1959): si consolida la professione del **programmatore** ed i progettisti/costruttori non sono più necessari per risolvere istanze di problemi.
- **Seconda generazione** (1959-1965): con i linguaggi di programmazione di alto livello gli **esperti dei problemi** sono coinvolti nel progetto del programma e ne diventano gli utilizzatori.
- **Terza generazione** (1965-1980): si diffonde l'industria del software, di conseguenza cala drasticamente la produzione di programmi costruiti ad hoc; i **clienti** sono **addestrati ad usare autonomamente sistemi** e programmi sviluppati dall'industria del software mentre i programmatori escono dai centri di calcolo delle aziende e tendono a concentrarsi nelle software house.
- **Quarta generazione** (dal 1980 in poi): con la nascita del **personal computer** l'**utente generico** è autonomo nel rivolgersi al mercato hardware e software; la gestione della sala macchine (nel caso di super computer) rimane affidata ad una struttura interna tradizionale; le software house offrono prodotti e servizi.

Seminari

Progetto SAGE:

Durante la guerra del pacifico (1942-1945) l'esercito americano ha grosse difficoltà nell'addestrare i piloti di aerei: i produttori di ogni tipologia di aereo fornisce insieme ad esso un simulatore analogico, il quale però spesso viene consegnato con molto ritardo non rendendo possibile per lungo tempo l'addestramento.

Nel 1944 l'ONR (Office for Naval Research, all'epoca nelle forze armate americane esisteva solo l'esercito e la marina) insieme al MIT decide di costruire un simulatore di volo universale, ovvero adattabile ad ogni tipo di aereo.

La direzione del progetto viene affidata a Jay **Forrester**, il quale in un primo momento tenta di costruire una macchina interamente analogica ma realizza subito che è troppo lenta ed imprecisa.

Di conseguenza nel 1945, dopo aver partecipato alla First Conference on Computing ed aver letto delle opere di Von Neumann, decide di utilizzare come cuore del simulatore un **computer digitale general purpose**.

Il 2 settembre del 1945 però la guerra col Giappone termina ed i fondi iniziano ad essere progressivamente ridotti, i lavori continuano ma più a rilento.

Nel 1947 inizia la progettazione della macchina che viene chiamata **Whirlwind**.

Whirlwind per poter essere utilizzato come simulatore deve essere **"real time"** (sistema dipendente dal tempo), di conseguenza deve essere anche molto veloce, per garantire ciò vengono introdotte due importanti innovazioni a livello hardware:

- Parallelismo interno: alcuni tipi di operazioni possono essere eseguite su più dati in parallelo (per esempio somme su più coppie di valori);
- memorie di ferrite: più veloci di quelle utilizzate fino a quel momento.

Nello stesso anno viene creata l'USAF (United States Air Force) come dipartimento a se stante con lo scopo di mettere a punto la teoria della "difesa strategica", ovvero individuare e bombardare in maniera preventiva le zone industriali e densamente popolate dei potenziali nemici.

Il 29 agosto del 1949 esplode la prima bomba atomica sovietica, ciò ha un grande impatto sul pubblico ed il governo americano decide sia di applicare la difesa strategica sia di sviluppare un piano di difesa passiva per proteggersi da eventuali attacchi.

Alla fine del 1949 si installa al MIT l'ADSEC (Air Defense Systems Engineering Committee) diretto da George **Valley** con lo scopo di elaborare un piano di difesa per il nord america.

Questo piano, presentato nel 1951, prevede una rete di radar fissi e mobili suddivisa in settori che coprono tutto il territorio nazionale.

Per elaborare i dati raccolti e reagire velocemente alle minacce devono essere utilizzati dei computer real time, di conseguenza l'attenzione di Valley si concentra su **Whirlwind** e sul **sistema DRR** (Digital Radar Relay) sviluppato dall'Air Force Cambridge Research Center per trasmettere i dati radar sfruttando la linea telefonica (viene inventato il modem).

Whirlwind quindi riceve di nuovo ampi finanziamenti dall'USAF nell'ambito di quello che viene poi chiamato progetto Charles.

Whirlwind adotta parole di 16/32 bit e l'aritmetica parallela (ma supporta solo numeri interi). Ha una memoria centrale a nuclei di ferrite con un tempo di accesso teorico di 6 μ s, mentre l'accesso effettivo alla memoria risulta alla fine di 8 μ s.

Raggiunge i 40 KIPS (Kilo Instructions Per Second) in media, con l'impiego di circa 5000 valvole.

Il tempo di un'addizione è 8 μ s, quello di una moltiplicazione 25,5 μ s, quello di una divisione 57 μ s.

Viene introdotto un "accumulatore" ovvero un registro di una parola presente nell'unità aritmetica che contiene uno dei due operandi e il risultato dell'operazione (che è molto spesso un operando dell'operazione successiva), riducendo così gli accessi alla memoria.

All'interno del progetto Charles vengono realizzati due prototipi dell'intero sistema:

- Cape Cod del 1953: serie di radar installati a Cape Cod collegati ad un computer Whirlwind I, lo scopo è raccogliere dati preliminari sul funzionamento delle apparecchiature.
- Cape Cod del 1954: viene ampliata la rete radar e migliorata la capacità di rendere grafica la traiettoria dei velivoli avvistati.

Si decide poi di procedere ad una sperimentazione più realistica cercando di simulare un intero "settore" (Progetto Claude).

Nel mentre vengono prodotte le specifiche per una versione aggiornata e molto più potente: Whirlwind II.

La costruzione di questa macchina, che presenta difficoltà maggiori, viene affidata nel 1952 alla **IBM** con il nome **AN/FSQ-7** (Army-Navy Fixed Special eQuipment).

Il prototipo viene consegnato nel 1955.

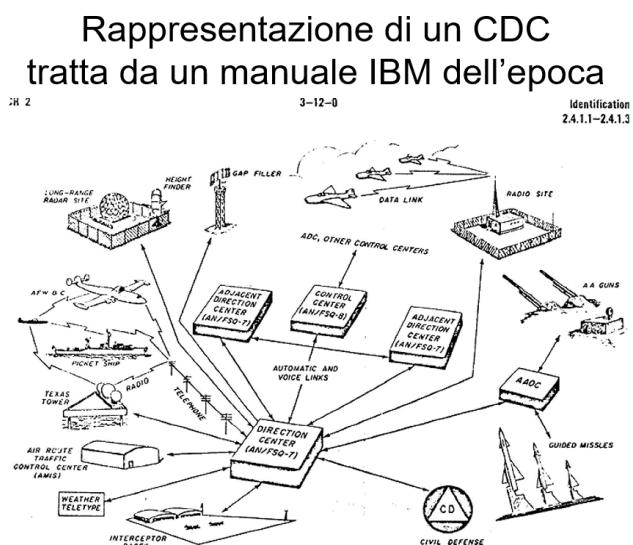
Sempre nel 1955 si passa all'ultima fase di sperimentazione a Lexington in Massachusetts. In questa occasione per indicare il sistema viene coniato l'acronimo **SAGE** (Semi Automatic Ground Environment) che diventerà il suo nome definitivo.

Un complesso sistema radar fornisce input simili in mole e tipo a quelli presumibili a regime, i dati vengono poi elaborati da un prototipo AN/FSQ-7 appena consegnato dalla IBM.

Inoltre un link radio connette il computer con i caccia per sperimentare la guida automatica.

Il sistema SAGE assume in via definitiva una struttura modulare con:

- Una serie di settori: ciascuno ha un CDC (Combat Direction Center) dotato di un computer (AN/FSQ-7) collegato alle installazioni radar, all'armamento di difesa ed ai settori contigui;
- Uno o più CCC (Combat Control Center) con computer (AN/FSQ-8, una versione più potente del precedente) collegato a tutti i CDC per il controllo e coordinamento delle operazioni.



AN/FSQ-7 ha circa 58000 valvole (alcune prodotte nell'est Europa) e 13000 transistor. Occupa 2000 metri quadri e pesa 275 tonnellate, inoltre usa circa un megawatt di potenza. Ogni sito ha due esemplari in hot standby in modo che se una macchina si guasta l'altra entra immediatamente in funzione.

Ha dei terminali CRT (a tubo catodico) con light gun (puntatore simile ad un pennino) collegata.

Il suo sistema operativo è chiamato PEC (Program Execution Control).

Il sistema pur essendo real time non ha interrupt, ma fa ampio uso di polling (verifica ciclica di tutte le unità o periferiche di input/output).

Le funzioni trigonometriche sono tabulate.

L'SDC (Systems Development Corporation, una società satellite della RAND, Research AND Development) si occupa della costruzione del software, mentre la Mitre corp (società satellite del MIT) si occupa della costruzione della rete di radar e computer.

Vengono individuati 22 settori in cui installare un CDC e 3 siti in cui installare un CCC; viene inoltre individuato un sito in Canada (Ontario) dove inserire un altro CDC.

Il primo sito viene completato nel 1958 (base McGuire nel New Jersey), mentre l'ultimo nel 1963.

Il servizio viene dismesso ufficialmente venti anni dopo nel 1983.

Il progetto SAGE lascia in eredità molte innovazioni in informatica, alcune di esse sono:

- **parallelismo** nella CPU,
- **memorie** a nuclei di ferrite,
- **hot standby** (high reliability),
- CRT, light gun, tecniche di presentazione grafica (J.C.R. Licklider lavorò sull'interfaccia e sulla comunicazione),
- modello **real time**,
- **modem**, uso di linee telefoniche, in generale telecomunicazione e networking,
- utilizzo di un **sistema operativo** (PEC),
- utilizzo del **polling**,
- **common area** (adottata da COBOL e altri OS),
- uno dei primi linguaggi "algebrici" (Jovial),
- tecniche di simulazione,
- **multiprogramming**.

Progetto SABRE:

Passando invece allo sviluppo dell'informatica per il pubblico e non più per scopi militari si hanno come progetti più importanti SABRE ed ERMA.

Col progetto SABRE nasce il sistema transazionale ed il concetto di database, infatti è un sistema per la gestione dei voli aerei (gestione dei posti liberi, gestione dei dati dei passeggeri, vendita dei biglietti).

Prima della seconda guerra mondiale per vendere i biglietti aerei si usano i seguenti metodi:

- negli anni '30: **request & reply**: ogni vendita dei biglietti veniva richiesta telefonicamente alla sede da cui partiva il volo, se c'era posto veniva dato l'ok.

- dal 1940 in poi: si vendono i biglietti e viene comunicato successivamente alla sede interessata (**sell & report**, operazione più veloce perchè non si attende una risposta) fino al raggiungimento di una soglia critica di riempimento dell'aereo (la sede incaricata tiene traccia dei posti occupati in una availability board), a quel punto la sede di partenza comunica il messaggio "**stop sale**" e si prosegue con request & reply fino al totale riempimento.

ma questo sistema è ancora lento e dopo la seconda guerra mondiale si dimostra incapace di gestire il traffico aereo, il quale nel mentre è notevolmente aumentato.

Un primo passo verso l'automatizzazione viene fatto dal **Reservisor**, ideato da Charles Amman nel 1946 e costruito dalla Teleregister Corporation.

Il suo scopo è quello di automatizzare la prenotazione di posti sugli aerei (mantiene solo il numero di posti liberi su un aereo, non gestisce il singolo posto a sedere).

È costituito da una matrice di relè (che sostituisce l'availability board) ed una serie di apparecchi da tavola che la possono interrogare, mentre l'inserimento delle nuove prenotazioni è più difficoltoso e richiede più tempo, per questo motivo viene utilizzata solo per automatizzare il segnale di stop sale (ottenibile interrogando la matrice).

Permette di servire 200 passeggeri al giorno in più con 20 impiegati in meno rispetto al metodo precedente.

Nel 1952 sempre Amman e la Teleregister Corporation presentano **Magnetronic**, una macchina in grado di gestire le prenotazioni e di mantenere i dati di ogni singolo posto a sedere.

Ha come memoria che sostituisce l'availability board un tamburo magnetico che può essere facilmente aggiornato.

Inoltre gli apparecchi per l'interrogazione sono più sofisticati (permettono anche la lettura di schede perforate) e permettono di fare interrogazioni più complesse.

Riesce a gestire fino a 2000 voli previsti per i successivi 31 giorni.

L'IBM invece nel 1956 realizza la **Reserwriter** basandosi su un card processor IBM 858.

Questa macchina oltre a gestire i singoli posti è in grado di mantenere anche i dati dei passeggeri.

La sua tecnologia si basa su schede perforate ed utilizza il teletype (trasmissione automatica di messaggi attraverso la rete telegrafica) per trasmettere le informazioni alle varie sedi.

Essendo però poco affidabile (8% di errori) è meno produttivo dei due modelli precedenti.

Nel 1953 L'**AA** (American Airlines) inizia a collaborare con l'**IBM** nel progetto **SABRE** (Semi Automatic Business Environment Research).

L'IBM è responsabile dell'hardware e del "control program" (sistema operativo), mentre l'AA è responsabile dello sviluppo applicativo e del deployment.

L'**AA** è quindi il **primo esempio di azienda privata che si dota di personale informatico**: 28 interni ed altri presi dall'esterno.

Questo gruppo di lavoro misto presenta il primo prototipo basato su una IBM 650 nel 1959 (ci vuole molto tempo perché i costi sono elevati ed i finanziamenti arrivano diluiti nel tempo).

Il sistema SABRE ha come principali funzionalità:

- gestione dell'inventario dello spazio di tutti i voli AA;
- prenotazione e cancellazione di tutti i posti da ogni ufficio AA;

- memorizzazione e controllo delle informazioni sui passeggeri e sui posti;
- richiesta di posti ad altre compagnie;
- gestione automatica dei ritardi, cambi di orario o di percorso;
- tempi di risposta minori o uguali a 3 secondi nel 90% dei casi.

Tra il 1960 ed il 1962 il sistema viene ulteriormente sviluppato.

Mel Perry (matematico che lavora al progetto) introduce la segmentazione in moduli di circa 500 istruzioni applicandola al programma di controllo IBM (sistema operativo di 18000 linee), questo migliora notevolmente le prestazioni e salva il progetto che altrimenti sarebbe fallito. Inoltre, ispirandosi al parallelismo di SAGE, viene introdotto il multitasking nel sistema operativo.

Dal 1962 al 1964 si ha il deployment negli uffici della AA, in questa fase risulta essenziale raddoppiare la memoria del calcolatore.

Le altre compagnie aeree osservando il successo di SABRE decidono di dotarsi anche loro di sistemi simili.

Nel 1965 l'IBM realizza **PARS** (Programmed Airline Reservation System) basato sul sistema 360 e su uno speciale sistema operativo chiamato ACP (Airline Control Program).

Il prodotto viene venduto alle compagnie Braniff, Continental, Delta, Northeast e Western, con installazioni a partire dal 1968.

La Eastern invece decide di collaborare con IBM per realizzare una versione di PARS con più funzionalità e prestazioni migliori, **Eastern-PARS** viene installato nel 1968.

United e la TWA invece decidono di costruire dei sistemi più avanzati di PARS appoggiandosi su altre aziende, ma entrambi i progetti falliscono ed adottano entrambe Eastern-PARS nel 1970.

Questo è il **primo caso** di acquisizioni di **sistemi informativi che vengono adattati** (e migliorati) **per altri clienti**.

Intorno alla metà degli anni '70 SABRE ha un'ulteriore evoluzione:

- viene trasferito su hardware più moderno (ultimi modelli dell'IBM 360),
- acquisisce nuove funzionalità (tariffe, emissioni di biglietti, ecc),
- migliorano i terminali (terminali video e stampanti veloci),
- ritorna competitivo con le varie versioni di PARS.

Data la coesistenza di vari sistemi si innesca una forte competizione che li porta ad evolversi fino a quelli che sono i sistemi moderni, attualmente i principali sono:

- Amadeus (evoluzione del sistema della Eastern),
- SABRE,
- Galileo (evoluzione di Apollo, ovvero la versione di Eastern-PARS della United),
- Worldspan.

Il progetto SABRE lascia inoltre in eredità all'informatica il **sistema transazionale** ed il **concetto di database**.

Progetto ERMA:

Il progetto ERMA invece riguarda il mondo bancario: nel 1950 negli USA sono emessi ogni giorno lavorativo più di 25 milioni di **assegni**, ognuno di questi assegni interessa da 1 a 3 banche e richiede una lavorazione di più di due giorni.

Infatti ogni assegno deve essere:

- **verificato** dalla banca che lo accetta: la verifica consiste nel raggruppare gli assegni per banca emittente riportando una distinta degli importi, alla fine della giornata i pacchi di assegni su altre banche sono inoltrati al circuito della Fed. Res. che provvede a consegnarli alla banca emittente;
- **contabilizzato** dalla banca da cui proviene: aggiornamento della scheda relativa al conto di provenienza.

Vengono già utilizzate delle macchine per smistare (fare il sort) gli assegni nella fase di verifica, ma sono poco efficienti.

Negli anni del dopoguerra negli USA si diffonde la preoccupazione che il problema del maneggio degli assegni sia un fattore limitante per lo sviluppo economico e finanziario.

Nell'aprile del 1950 Hoot Gibson della **SRI** (Stanford Research Institute) incontra Clark Beise della **BofA** (Bank of America, la più grande banca del mondo) e decidono di cercare un modo per automatizzare la procedura degli assegni.

Si realizza subito che occorre cambiare la classificazione dei conti: non è più solo sufficiente il nome dell'intestatario ma è necessario assegnargli un **numero progressivo**.

Inoltre si cerca un modo di rendere gli assegni leggibili da una macchina, scartata la proposta di utilizzare inchiostro fluorescente (che risulta essere facilmente riscrivibile) si decide di utilizzare inchiostro magnetico (il cui processo di lettura è detto MICR).

Viene quindi progettata la macchina **ERM** (Electronic Recording Machine), in grado di lavorare alla massima efficienza maneggiando 30000 conti e servendo 12 filiali.

Nessuna azienda la vuole realizzare perché la stima dei costi è troppo alta, di conseguenza la SRI realizza il **primo prototipo** in proprio ed i lavori terminano nel 1955, in questa occasione il nome viene cambiato in **ERMA** (Electronic Recording Machine Accounting).

Questo prototipo ha:

- 8200 valvole (i transistor erano ancora troppo poco affidabili), 34000 diodi, 500 relè, due tamburi magnetici e 12 nastri.
- un peso totale (compreso le unità di I/O) di 25 tonnellate,
- un programma "wired", quindi non in memoria ma modificabile cambiando i collegamenti tra le varie parti.

Il 22 settembre 1955 il sistema ERMA viene presentato al New York Times e ad altri giornali, a questo punto le aziende costruttrici di macchine mostrano di nuovo interesse.

Viene scelta come costruttrice la **General Electric** (GE) e nel 1956 inizia la produzione.

L'installazione delle macchine avviene tra il 1959 ed il 1961, mentre l'inaugurazione viene fatta nel 1960.

La conversione ai nuovi conti con un numero di serie dura fino al 1962.

La **nuova macchina** prodotta dalla GE è però **completamente diversa** dal prototipo in quanto sono passati diversi anni.

Essa ha:

- 14000 transistor, 45000 diodi, clock di 250 kilohertz,
- memoria a nuclei di ferrite di 4000 parole da 56 bit (14 caratteri di 4 bit),
- 8 nastri magnetici,
- programma memorizzato: sistema operativo e applicazione,
- stampanti veloci (750 righe al minuto),
- connessione con il sorter per preparare l'input alle elaborazioni successive.

Il sorter è il dispositivo che ordina gli assegni (leggendo con MICR) alla velocità di 10 al secondo e, per scartare gli assegni non validi (detti fermati), deve essere connesso al computer (in cui si trova l'elenco).

Anche il modo di operare è completamente diverso da quello del prototipo.

Infatti gli assegni vengono ordinati dal sorter connesso ad ERMA (che comanda l'esclusione di quelli "fermati" e li memorizza su nastro).

Mentre avviene il sorting ERMA produce un nastro "del giorno" ordinato, questo viene elaborato con il nastro dei conti (old master) per produrne uno nuovo (new master) con i conti aggiornati.

Contemporaneamente sono prodotte tutte le "scritture" necessarie sull'assegno e sulla modulistica.

Questo sistema dà un grosso vantaggio competitivo a BofA che grazie ad esso in futuro sarà la prima a dotarsi di carte di credito.

Domande da esame

Di seguito sono riportate alcune domande che il professore ha indicato essere simili ad alcune che potrebbe fare all'esame.

La risposta riportata (se presente) è all'incirca quella fornita dal professore.

Dove e come compare la scrittura?

Quale è la principale differenza tra la tavoletta amministrativa del sito di Fara e le leggi di Hammurabi?

Le leggi di Hammurabi descrivono come l'individuo specializzato in un determinato campo deve procedere (e anche come devono essere gestiti i casi di errore!), cosa che non accade nella tavoletta.

Quindi le leggi di Hammurabi descrivono per la prima volta come deve essere fatto un processo.

La grammatica di Panini in un altro contesto rappresenta la stessa rivoluzione, indicando il processo per costruire un linguaggio.

Stessa cosa in Cina con l'algoritmo dell'I Ching per interpretare i risultati dell'oracolo.

In cosa evolvono la grammatica di Panini e gli oracoli cinesi?

La grammatica di Panini evolverà fino ai calcolatori, mentre gli oracoli cinesi ispirano Leibniz. Leibniz vedendo che i cinesi avevano sviluppato un sistema parziale per interpretare gli oroscopi che funzionava, capisce che è sufficiente un sistema binario per rappresentare un qualunque numero.

Perché nasce la retorica?

Perché in Grecia con la nascita della democrazia era necessario trovare un modo di convincere gli elettori o una giuria in un tribunale.

Da qua la ricerca di un metodo sempre più cogente fa nascere la logica.

Come evolve la storia del sillogismo?

La storia del sillogismo parte da Aristotele e si evolve nel medioevo fino a terminare con 24 sillogismi validi ovvero con un significato cogente (dai 256 di partenza). (è da sapere come funzionano i sillogismi!).

Come si classifica il sistema di Euclide?

il sistema formale di Euclide non è universale: funziona solo per il mondo della geometria, è invece generale in quanto applicabile a (quasi) tutti i problemi di geometria.

Qual'è la differenza tra un sistema di numerazione e un sistema di scrittura?

Quali sono i passi fondamentali dei sistemi di numerazione?

Disaccoppiare il numero dall'oggetto contato:

I numeri devono essere interpretati in maniera univoca come cardinalità di un qualunque insieme di elementi, non più come cardinalità di un insieme di oggetti (quindi 3, non 3 pecore) e ciò è avvenuto con l'avvento delle bulle e poi con le tacche sulle tavolette.

Scoperta dello 0:

Prodotto dell'aritmetica indiana, gli europei impiegano tantissimo tempo ad adottare questo simbolo e questo concetto.

Perché la proposta di Leibniz prende piede e viene seguita mentre quelle di Lullo ed Hobbes no?

I sistemi di Lullo ed Hobbes tentavano di risolvere il problema disciplinando il linguaggio naturale.

Leibniz invece capisce che bisogna trovare un linguaggio nuovo che possa ridurre ogni argomentazione ad un calcolo, questo linguaggio deve avere dei simboli e delle operazioni per manipolarli in modo da ottenere un risultato.

Che cosa calcola il linguaggio di Boole?

Come termina l'evoluzione di questo linguaggio formale per descrivere tutte le argomentazioni?

Termina con la macchina di Turing, in quanto è il linguaggio più semplice che descrive tutte le funzioni calcolabili (ipotesi di Church-Turing), in altre parole con questo linguaggio sono calcolabili tutti i problemi calcolabili.

Quindi dal punto di vista matematico il problema è risolto.

Dal punto di vista pratico sarà Von Neumann a modificare il linguaggio di Turing ed a crearne un altro in grado di funzionare efficacemente su un dispositivo operativo non umano.

Quale è una grossa rivoluzione di Babbage?

Fino a quel momento i dispositivi operativi erano stati esseri umani via via sempre più specializzati, da questo momento in poi nascono dispositivi operativi che sono delle macchine.

Che cosa è successo di importante nel periodo che intercorre tra l'ENIAC e l'EDVAC?

È nata la cibernetica, la quale ha permesso di unire tutte le discipline che danno poi origine all'informatica, in particolare lo sviluppo dei dispositivi operativi e linguistici.

La cibernetica nasce in seguito alla scoperta dei neuroni.

Shannon contribuisce alla nascita di questa disciplina perché introduce calcolatori elettronici più efficienti di quelli analogici.

Turing contribuisce dimostrando che tutto il calcolabile si può calcolare con calcolatore, quindi il tentativo di emulare i neuroni è fattibile.

Von Neumann contribuisce col suo linguaggio e con la sua macchina operativa.

Da dove parte la storia dell'informatica?

Dal big bang, in quanto fin dalla sua nascita l'universo codifica informazioni al suo interno.

Quante sono le possibili interpretazioni di una frase in linguaggio naturale?

Ce n'è almeno una per ascoltatore/lettore.

Qual è la grossa innovazione dell'ENIAC?

È la prima macchina che dimostra la superiorità dell'elettronica sulle macchine analogiche.

L'elettronica è importante perché permette di avere prestazioni maggiori e soprattutto perché permette di avere memoria riscrivibile, quindi si possono modificare parti del programma in esecuzione.

Quali sono i contributi dei progetti di cui ha parlato Teolis nei seminari (SAGE, SABRE, ERMA)?

Il computer non viene più utilizzato per produrre solo tabelle, ma si sviluppano tutte quelle applicazioni del data management, inoltre si sviluppa il connubio tra informatica e telecomunicazioni.

Degni di particolare nota sono lo sviluppo grazie a quei progetti di sistemi in tempo reale, database e sistemi transazionali.

Come si colloca il logicismo nella storia dell'informatica?

Il logicismo nasce dalla proposta di Frege di creare una logica matematica con la quale dedurre tutte le teorie.

I matematici nel tentativo di fare questa cosa definiscono il teorema di Hilbert, ovvero che ogni teoria è deducibile da un determinato sistema in un tempo finito.

Questo spinge Turing a definire la sua macchina universale che confuta il teorema.

Inoltre il sistema formale di Frege (che ha generato il logicismo) è un primo linguaggio di programmazione, al punto tale che PROLOG è stato creato usando il suo sistema formale.

Che ruolo ha la cibernetica nello sviluppo dell'informatica?

La cibernetica è in grado di far toccare la teoria di Shannon sull'elettronica, la logica di Boole, i teoremi sul campionamento dell'elettronica, le teorie di Von Neumann.

Inoltre fornisce lo stimolo per la costruzione di un manufatto che simuli la capacità cognitiva dell'essere umano, quindi si passa da calcolatori di tavole a computer general purpose.

Che la proposta di Von Neumann abbia come idea quella di avere un cervello elettronico è visibile dalla nomenclatura utilizzata: computer (fino a quel momento il computer era una persona, una specie di contabile), memoria, ecc... ovvero tutte parole che riguardano le cose vive e non macchine.

Che cosa ha permesso a Babbage di arrivare all'idea della macchina per le differenze?

La necessità di costruire tavole con meno errori possibili.

Esisteva il progetto di De Prony che dimostrava che per costruire le tavole erano sufficienti persone che sapessero fare delle somme (e prima ancora Smith che aveva diviso il lavoro in fasi).

Jacquard aveva inventato il telaio meccanico, dimostrando che si potevano fare macchine che eseguissero delle operazioni.

Perché nasce il sillogismo?

Abbandonate le spiegazioni fornite dal mito e dalla religione è nata la necessità di organizzare la società in maniera complessa, con la conseguente necessità di convincere delle giurie di una determinata tesi.

Per questo motivo nascono prima la dialettica per convincere i propri pari e poi la retorica per convincere le masse.

Da qui nasce la necessità di avere argomentazioni sempre più cogenti e per questo motivo nasce la logica.

Il primo ad applicare la logica alle argomentazioni in maniera formale è Aristotele col sillogismo.

Il sillogismo è sempre un linguaggio naturale disciplinato, non si utilizza ancora un linguaggio formale, ciò accadrà solo con Leibniz che abbandona definitivamente il linguaggio naturale.

Perché Eckert e Mauchly non sono considerabili padri del computer come Von Neumann, nonostante abbiano avuto anche loro l'idea di mettere il programma nella memoria?

Avere il programma in memoria permette di modificare il programma stesso (fare salti, loop, ecc... questo non era possibile con le schede perforate che andavano lette in sequenza), questo porta Von Neumann a progettare anche il linguaggio di programmazione della macchina.

Von Neumann è il primo a progettare un linguaggio di programmazione ed a definire quali istruzioni deve avere necessariamente un linguaggio (istruzioni aritmetiche, istruzioni per essere turing completo, istruzioni che cambiano le istruzioni).

Eckert e Mauchly invece non considerano l'aspetto linguistico e non comprendono le potenzialità di una macchina con memoria elettronica e riscrivibile.