Perspectives in Computer and Software Technology
Perspektiv på data- och mjukvaruteknik

TDDE25

Fö 1
Course Introduction
History of Computing

IDA Page: https://www.ida.liu.se/~TDDE25/index.en.shtml

LISAM Page: https://liuonline.sharepoint.com/sites/Lisam_TDDE25_2022HT_NI

LINKÖPING UNIVERSITY

Examinator, kursledare: Jonas Kvarnström

Tar över kursen i år!

Course Themes

Perspectives in Computer and Software Technology

History of Computing

Computer Systems

Bits, Bytes & Numbers

Networking

Operating Systems

Internet

Algorithms

Data Abstractions

Programming Languages

Software Engineering

Databases

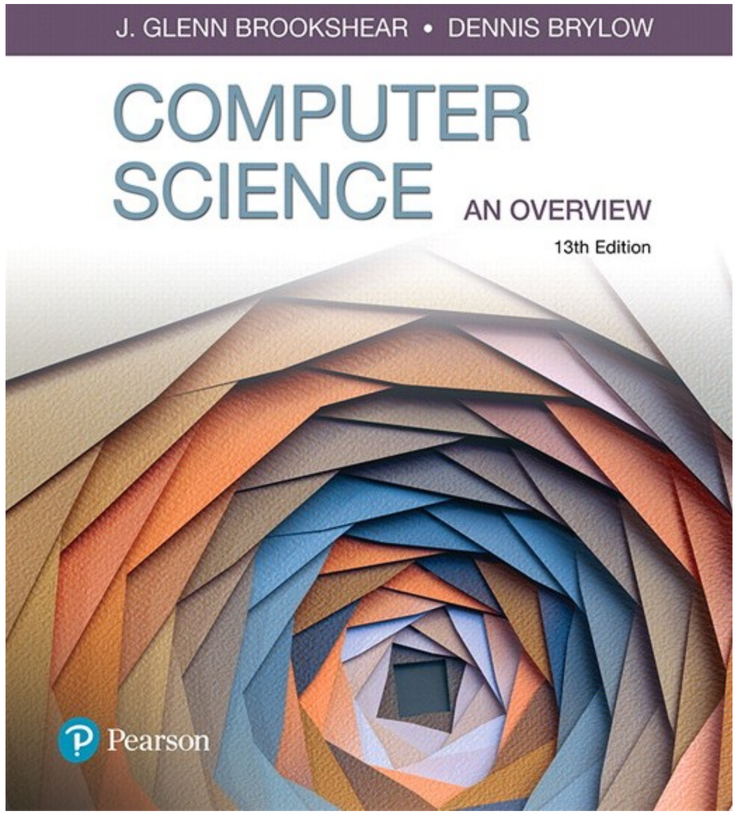
Artificial Intelligence

Sustainability

Theory of Computation

Course: Period 1

- Period 1: **15 seminars**
  - By knowledgeable researchers
    who **specialize** in each of the areas
  - **Lectures and books** complement each other:
    Read the **chapters** specified on the web
    before each seminar!
  - The book may be available in the bookstore,
    or can be ordered online (physical / e-book)

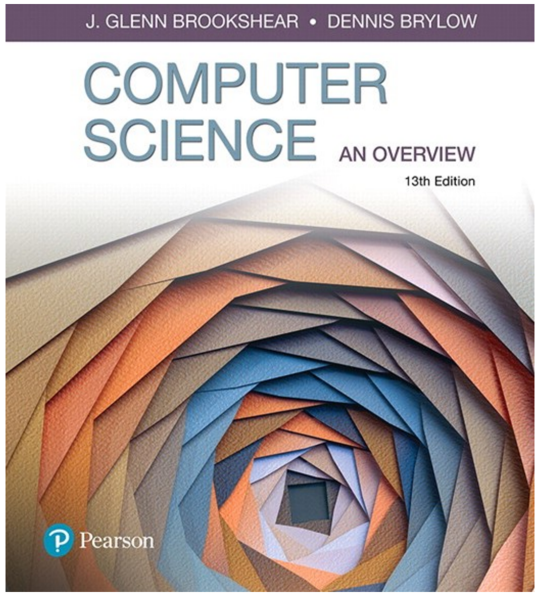


LINKÖPING UNIVERSITY

Why do we study this?

- **Provide perspectives for your entire education!**
  - What **is** computer science / computer engineering?
  - What will you **learn more about** during the next 5 years?
  - How do the **other courses** fit in?
  - What **connections** exist between different topics?

- ...and also, there will be an examination...
  - ...but the **main purpose** of the book + lectures is to learn more *for your future studies (and beyond)!*



Examination, period 1: Quizzes

- Most seminars (12 of 15, not this one) have a **10-15 minute quiz**
  - Distributed at random times during the seminar;
    can not be taken at another time!
  - Should not be too difficult if you read the book, follow the lectures
  - Must pass **9 out of 12**.
    If not: Write a 10 page report on a topic we agree on, related to the course
    (can't take a quiz later, even if you were sick)
  - Details on the IDA course web site under "examination".
  - Quizzes will be found at the LISAM course web site:
    - https://liuonline.sharepoint.com/sites/Lisam_TDDE25_2022HT_NI
    - Quiz registration: automatic (if you are registered for the course)

Course: Period 2

- Period II: **<u>Programming Project</u>** in Python (4 hp)
  - Groups of 3 students
  - Details will be presented in another seminar

- What does **<u>this</u>** have to do with the **<u>course book</u>** and **<u>quizzes</u>**?
  - In many ways, more closely related to TDDE23-24…
  - Essentially, most courses are 6 points;
    the programming project is a *separate aspect* of this course
  - Important for your programming and collaboration skills…
    and for **<u>perspectives on programming!</u>**

Computation: Some Fundamental Questions

What is Computation?

What is Computing?

What is a Computer?

What is Computation? Wikipedia...

- Computation is any type of calculation or the use of computer technology in Information Processing. [Application oriented]

- Computation is a process following a well-defined model understood and expressed in an algorithm, protocol, network topology, etc. [Algorithmics]

- Computation is also a major subject matter of computer science: it investigates what can or cannot be done in a computational manner. [Computability]

Computation as a Physical Phenomenon

A computation can be seen as a **purely physical** phenomenon occurring inside a closed physical system called a computer.

Examples of such physical systems include digital computers, mechanical computers, quantum computers, DNA computers, molecular computers, analog computers or wetware computers.

This point of view is adopted by a branch of theoretical physics: The physics of computation.

An even more radical point of view is the postulate of digital physics (Fredkin) that the evolution of the universe itself is a computation - Pancomputationalism.

Computing Curricula

Curriculum: *the **subjects** comprising a course of study in a school or college – what is important?*

- The Association for Computing Machinery (**ACM**) and **IEEE Computer Society** jointly sponsor the development of **Computing Curricula**
- Offer guidelines for undergraduate courses in computing

IEEE COMPUTER SOCIETY

Association for Computing Machinery

What is Computing?

ACM Computing Curricula 2005: **A Broad View**

"In a general way, we can define **computing** to mean any goal-oriented activity ***requiring, benefiting from, or creating*** computers.
Thus, computing includes:

- designing and building **hardware and software systems** for a wide range of purposes;
- processing, structuring, and managing various kinds of **information**;
- doing **scientific studies** using computers;
- making computer systems **behave intelligently**;
- creating and using **communications and entertainment media**;
- **finding and gathering information** relevant to any particular purpose, **and so on**.

The list is virtually endless, and the possibilities are vast."

What is Computing?

1989 ACM report on Computing as a Discipline[2]: **A Narrow View**

- The discipline of computing is the systematic study of **algorithmic processes** that **describe and transform information**: their theory, analysis, design, efficiency, implementation, and application. The **fundamental question** underlying all computing is "**What can be (efficiently) automated?**"

What is a Computer?

- The first use of the word "computer" was recorded in 1613:
  A **person** who carried out calculations / computations

- Kept its meaning until the middle of the 20th century;
  after that: a machine that carries out computations.

A computer is a general purpose device that can be programmed to carry out a finite set of arithmetic or logical operations. Since a sequence of operations can be readily changed, the computer can solve more than one kind of problem.

Computing Curricula: CC2020

- The newest ACM/IEEE **computing curriculum overview report** is CC2020
  - Moves from **Knowledge-based** Learning to **Competency-based** Learning
  - General agreement that career success requires **three things**:
    - **Knowledge – "Know what"** – A proficiency in core concepts and content and the application of learning to new situations
    - **Skills – "Know how"** – the ability to carry out tasks with determined results
    - **Dispositions – "Know why"** -- intellectual, social, or moral tendencies.
  - Any definition of competency must connect these three dimensions:

Competency = Knowledge + Skills + Dispositions

CC2020

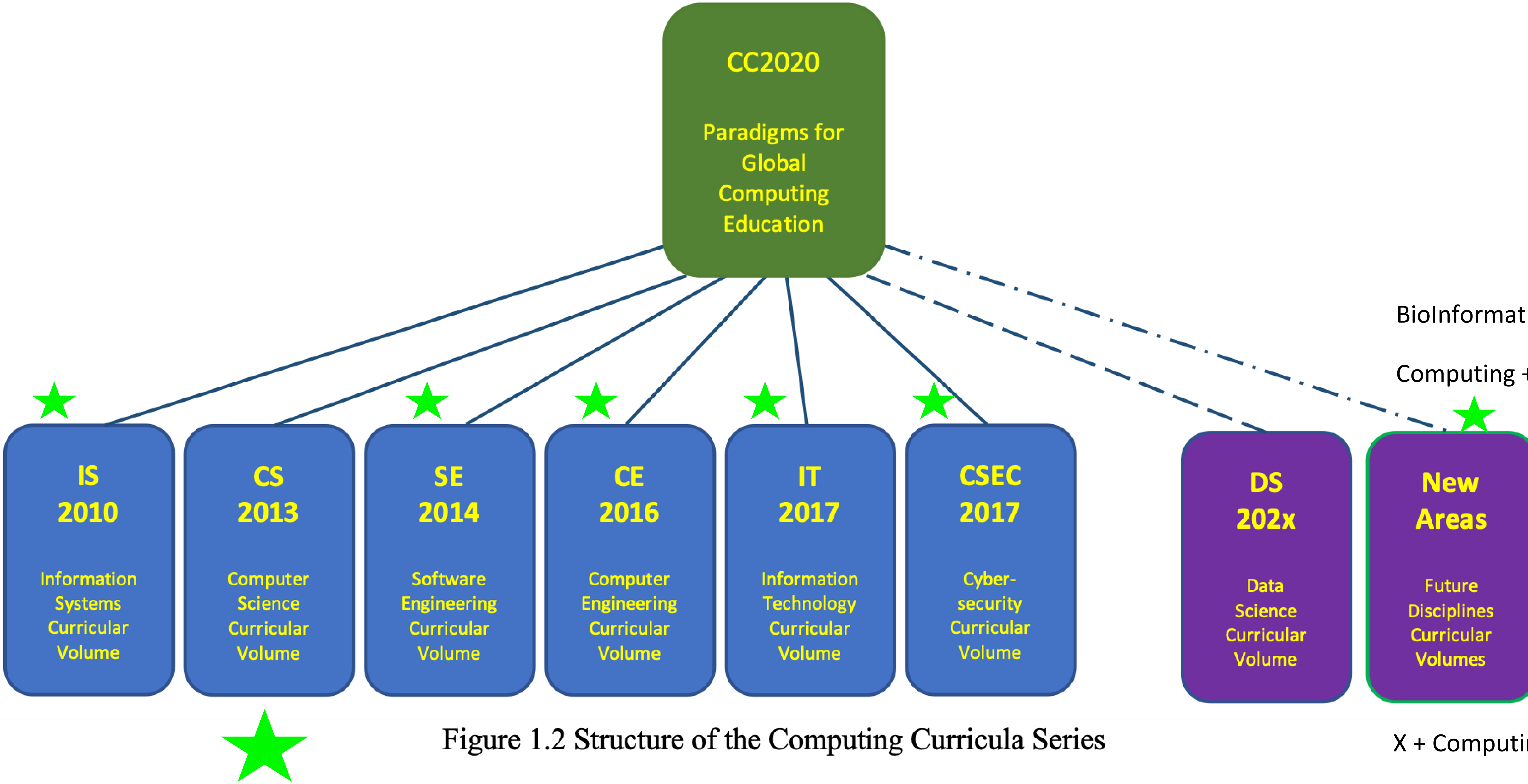


Figure 1.2 Structure of the Computing Curricula Series

BioInformatics

Computing + X

X + Computing

Computational Finance

CS2013: 18 Knowledge Areas

Even Computer Science is **not** just about programming!

- ★ AR - Architecture and Organization
- ★ CN - Computational Science
- ★ DS - Discrete Structures
- GV - Graphics and Visual Computing
- HC - Human-Computer Interaction
- ☆ IAS - Information Assurance and Security
- ★ IM - Information Management
- ★ IS - Intelligent Systems
- ★ NC - Networking and Communications
- ★ OS - Operating Systems
- ☆ PBD - Platform-based Development
- ☆ PD - Parallel and Distributed Computing
- ★ PL - Programming Languages
- ★ SDF - Software Development Fundamentals
- ★ SE - Software Engineering
- ★ SF - Systems Fundamentals
- ☆ SP - Social and Professional Issues

- PBD - New types of platform specific programming environments such as the web or mobile devices.
- PD - Now consolidates these topics in one area
- SDF - The entire software development process
- SF - Unified systems perspective for computing systems

★ Covered

☆ Less covered

Scheduled Seminars

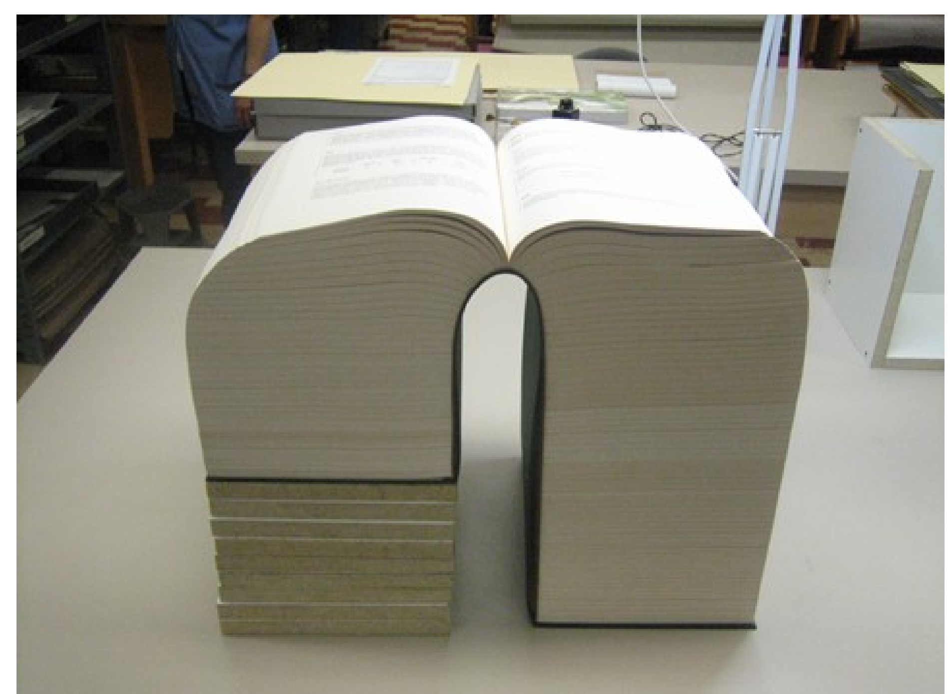
- Computer Systems
  - 1: Introduction & History of Computing (ch0) [Jonas Kvarnström]
  - 2: Data Storage/Number Systems (ch1) [Jonas Kvarnström]
  - 3,4: Data Manipulation/Computer Systems (ch2) [Petru Eles]
- Programs and Processes
  - 5,6: Networking and the Internet (ch4) [Niclas Carlsson]
  - 7,8: Algorithms/Computability (ch5, 12) [Victor Lagerkvist, Peter Jonsson]
  - 9: Software Project Descriptions [Jonas Kvarnström, Cyrille Berger]
  - 10: Database Technologies (ch9) [Olaf Hartig]
  - 11,12: Operating Systems (ch3) [Christoph Kessler]
  - 13: Software Engineering [Kristian Sandahl, Niklas Lanzen (Ericsson)] (ch7)
- Applications and Use
  - 14: Sustainability (—) [Anders Jidesjö]
  - 15: Artificial Intelligence (ch11) [Jonas Kvarnström]

History of Computing

Some Highlights!

Partly a lecture…

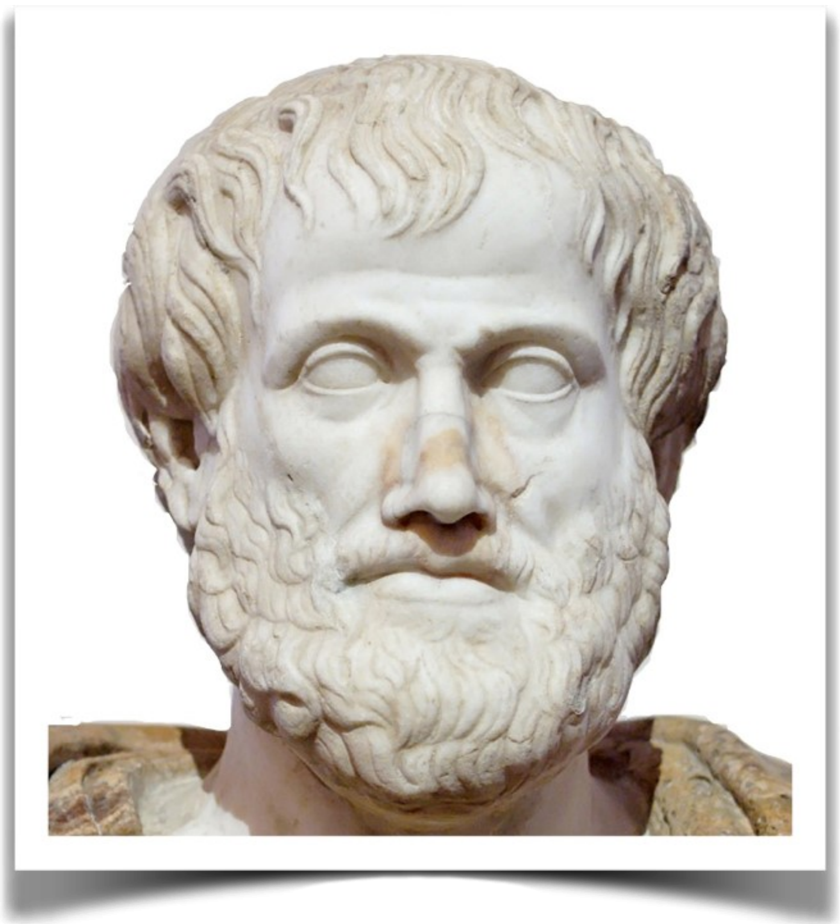
Partly reference materials!



LINKÖPING UNIVERSITY

Socrates
Plato
Aristotle

Aristotle and the Greeks



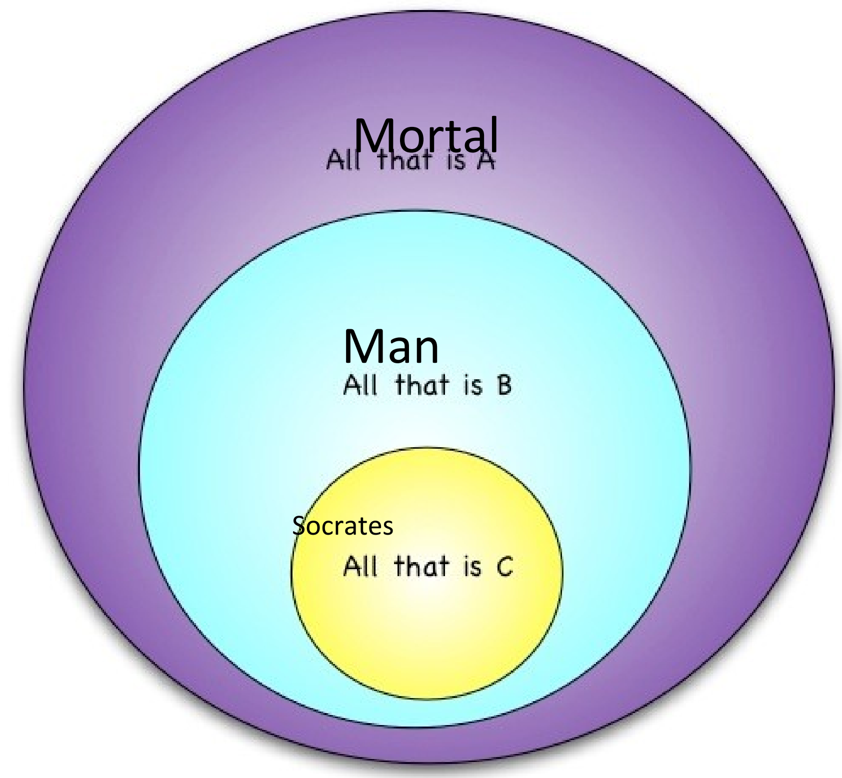
What is a good argument?

Origins of Computation begin with Reasoning!

Formalizing mental Processes

Syllogistic Reasoning

Syllogistic reasoning is a type of deductive argument. It involves trying to categorize objects by fitting them into contained circles. For instance, suppose we know that all the things matching category "B" fits completely inside the larger category of "A." That's our "major premise" or our first argument. Suppose we also can prove that all the things matching category "C" also fit inside the category of "B." That's our"minor premise" or our second argument. From these two statements, we can also conclude that all of "C" must fit in category "A" as well. We can see this if we chart it visually with three circles, like the drawing below.



All humans are mortal — Major Premise

Socrates is a human — Minor Premise

____

Socrates is mortal — Deductive Conclusion

Mechanical Roots of Computation

As far back as Aristotle and Plato:

Recreate human mental and physical processes using available technology

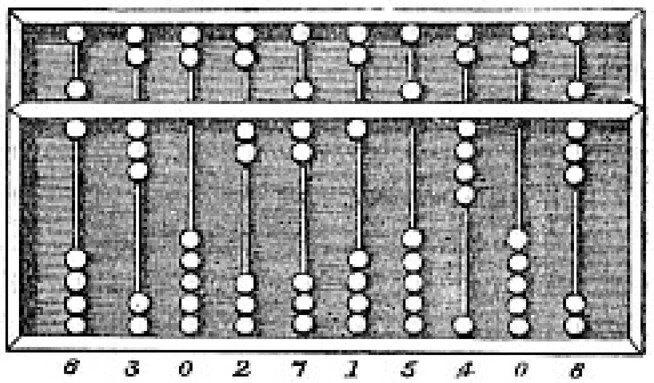
Mechanical Techniques

Electro-Mechanical Techniques

Electronics

The Abacus

An Abacus is a calculating tool for performing arithmetic processes



Chinese Abacus - Suanpan
both decimal/hexadecimal
computation
1st written mention 100 AD

Calculating Table
1508 Europe

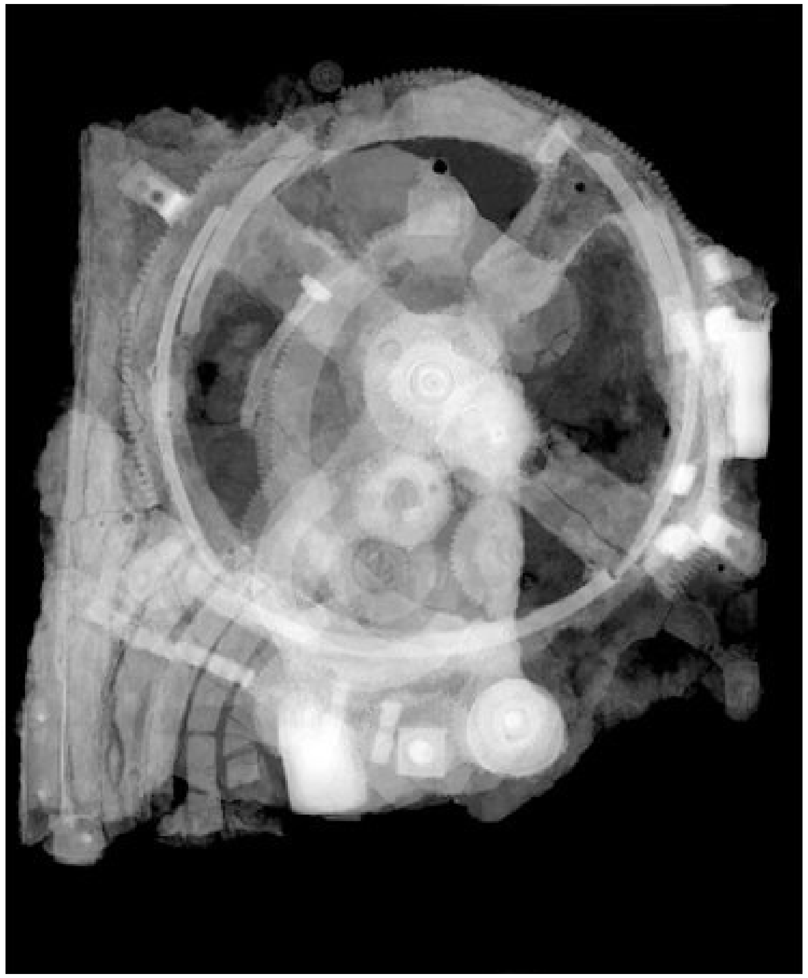
Modern abacus

The idea dates back as far as Babylonia (2400 BC)
Still in use today in some parts of the world

Antikythera Mechanism

Over 2000 years old...

Discovered in a shipwreck off the greek island of Antikythera in 1901

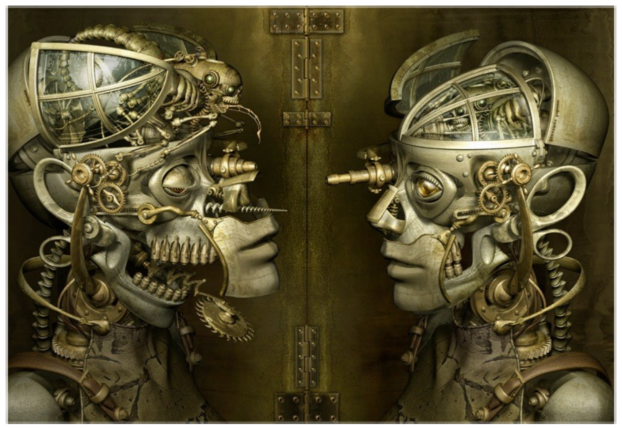


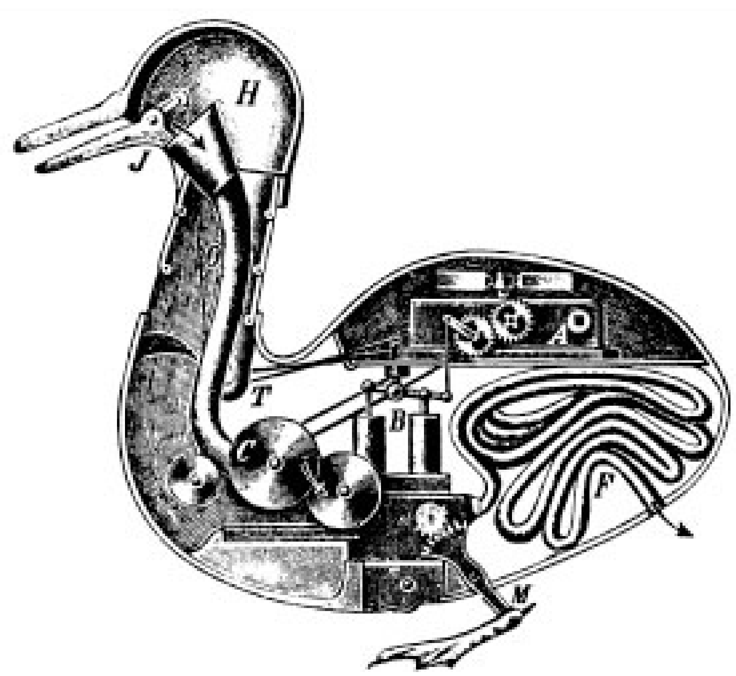
- The oldest known scientific calculator.
- Complex arrangement of over 30 gears
- Could determine with remarkable precision the position of the sun, moon and planets, predict eclipses and track the dates of Olympic Games.

LINKÖPING UNIVERSITY

Automatons - Precursors to Robotics

Natural Laws are capable of describing/producing complex behavior
Perhaps these laws govern human behavior?





16th Century onwards

Napiers Bones [1617]

Reduced multiplication and division to a series of additions and subtractions. Used a table-based calculation method that probably originated in India in the Middle Ages.

John Napier (1550 - 1617)

Also invented:
natural logarithms and
decimal notation

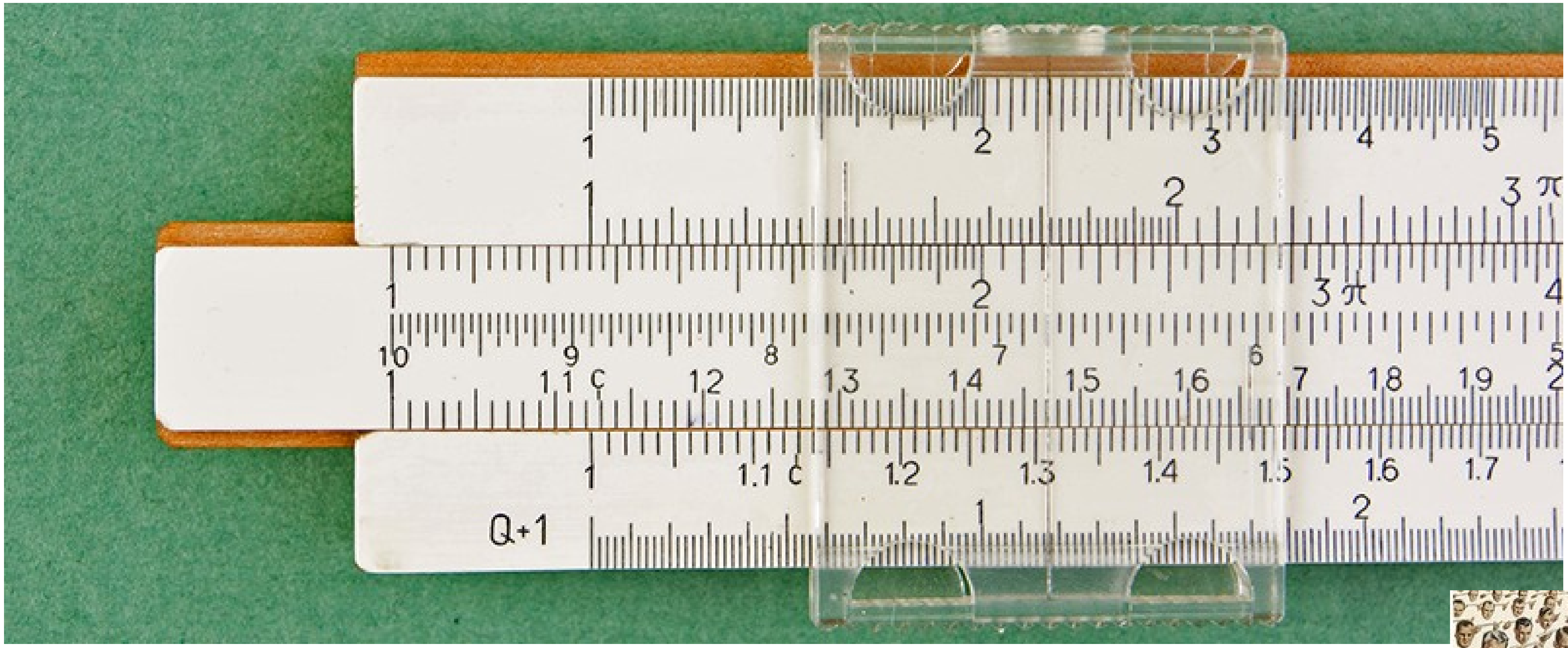
Users arranged rows of “bones” (bars inscribed with number sequences), then added certain numbers on the bars to do multiplication.

Slide Rule [1620]

William Oughtred
Slide rule (1620)

Straightforward principle:
Two bars, each marked with scales, slide next to each other.
Aligning numbers on different kinds of scales allows different calculations, such as multiplication or trigonometry.
Accuracy is limited and depends on the user's skill.

Slide Rule (later)



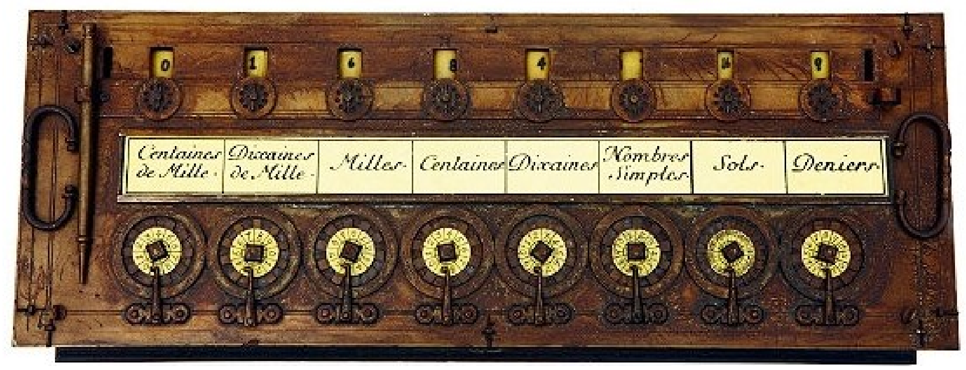
IBM advertisement
1949 - Card programmed electronic calculator

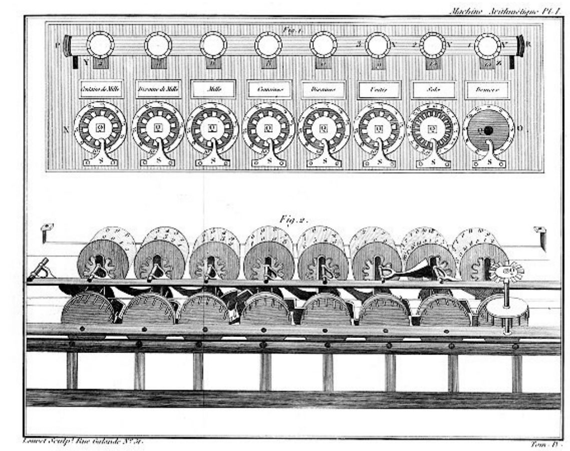
Pascal: The Pascaline [1642]

The world's first <u>automatic</u> calculating machine!

For addition and subtraction, the "<u>algorithm</u>" was performed by the machine and not by the human using the machine!

Blaise Pascal
1623 - 1662





1st mass produced commercial calculating machine
built 50, sold 15
(*too unreliable due to mechanical problems*).

Leibniz: The Stepped Reckoner [1673]

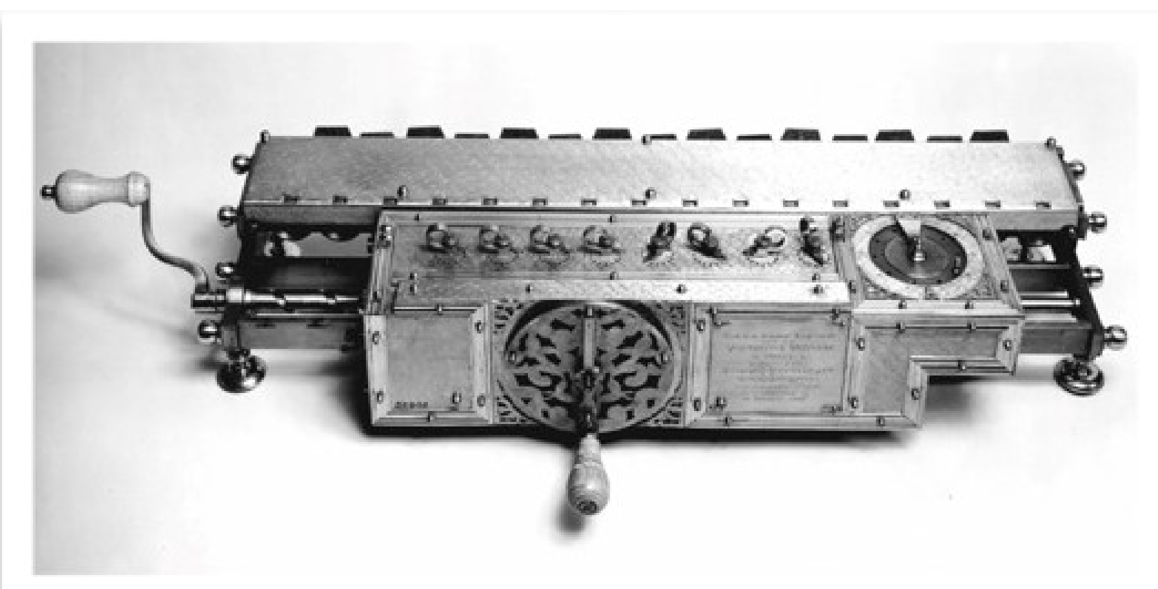
Leibniz (1646 - 1716)

Calculus Ratiocinator

- A universal artificial **mathematical language**
- Idea: All human knowledge could be represented in this language
- Calculational rules would reveal all logical relationships among these propositions
- Machines would be capable of carrying out such calculations

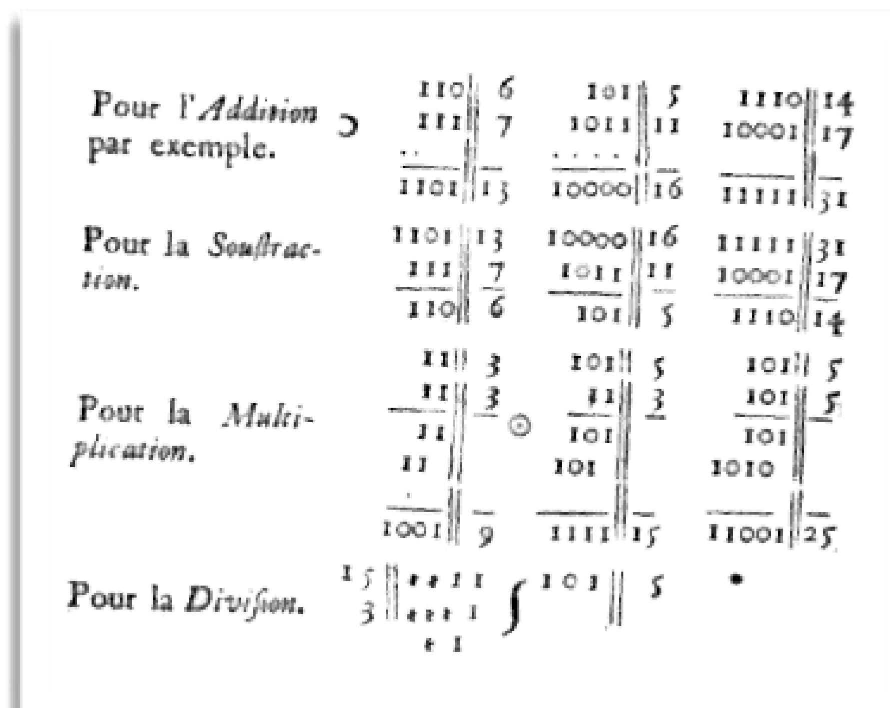
Let us Calculate!

In 1673, Leibniz built the first true four-function calculator. His unique, drum-shaped gears formed the basis of many successful calculator designs for the next 275 years, an unbroken record for a single underlying calculator mechanism.



Leibniz Step Reckoner

Addition
Subtraction
Multiplication
Square root extraction



Early use of binary system (not in step reckoner)

Babbage: The Difference Machine [1821]

*"One evening I was sitting in the rooms of the Analytical Society at Cambridge....
with a table of logarithms lying open before me. Another member, coming into the room,
and seeing me half asleep called out, "Well Babbage, what are you dreaming about?"
to which I replied, "I am thinking that all these tables might be calculated by machinery."*

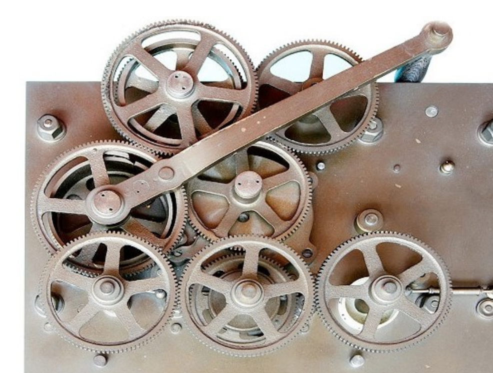
Charles Babbage
(1791-1871)

From 1821 to 1833, Babbage worked on a Difference Engine to produce accurate tables. This machine was at the edge of what was technically feasible at the time.

Due to problems with technology , politics and financing, it was never built to completion. (But it worked!)

This first difference engine would have been composed of around 25,000 parts, weigh fifteen tons (13,600 kg), and would have been 8 ft (2.4 m) tall.

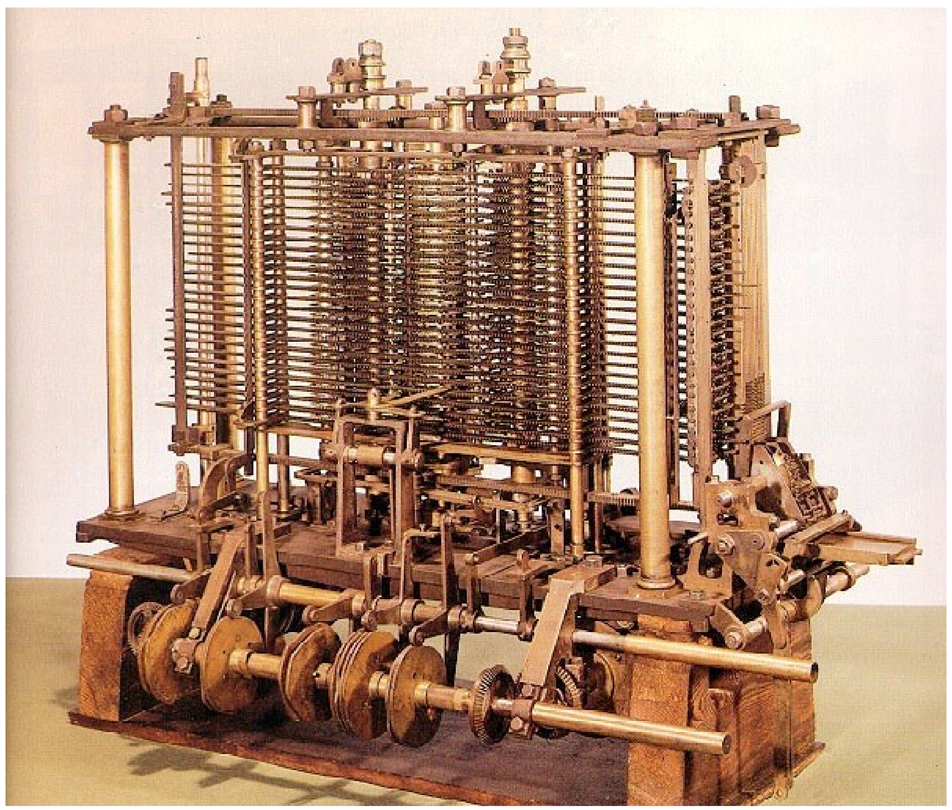
Babbage Difference Engine No. 1
3/4 -scale replica



Babbage: The Analytical Engine

Rather than a machine which could only perform specific computations, Babbage had a far greater idea called the Analytical Engine:

Construct a machine that could be programmed to solve any possible logical or computational problem!



The "mill" or computing part of the Engine
In modern terms: Central Processor (CPU)
Had registers for performing both logical and arithmetical computations

The punch card reading system for reading and storing data and programs in memory. Included a machine language very similar to modern ones.

It had random access memory of 1000 words of 50 decimal digits each (Equiv to 175,000 bits!)

Also included a printer for producing charts and images from the computations

Babbage's Analytical Engine was never fully built due to cost overruns and the inventor's cranky personality. Study of the designs shows that the system would have worked, and would have been comparable to mechanical computers built 100 years later at the end of the WWII.

Ada Lovelace: The 1st Programmer

Regarded as the word's first computer programmer!

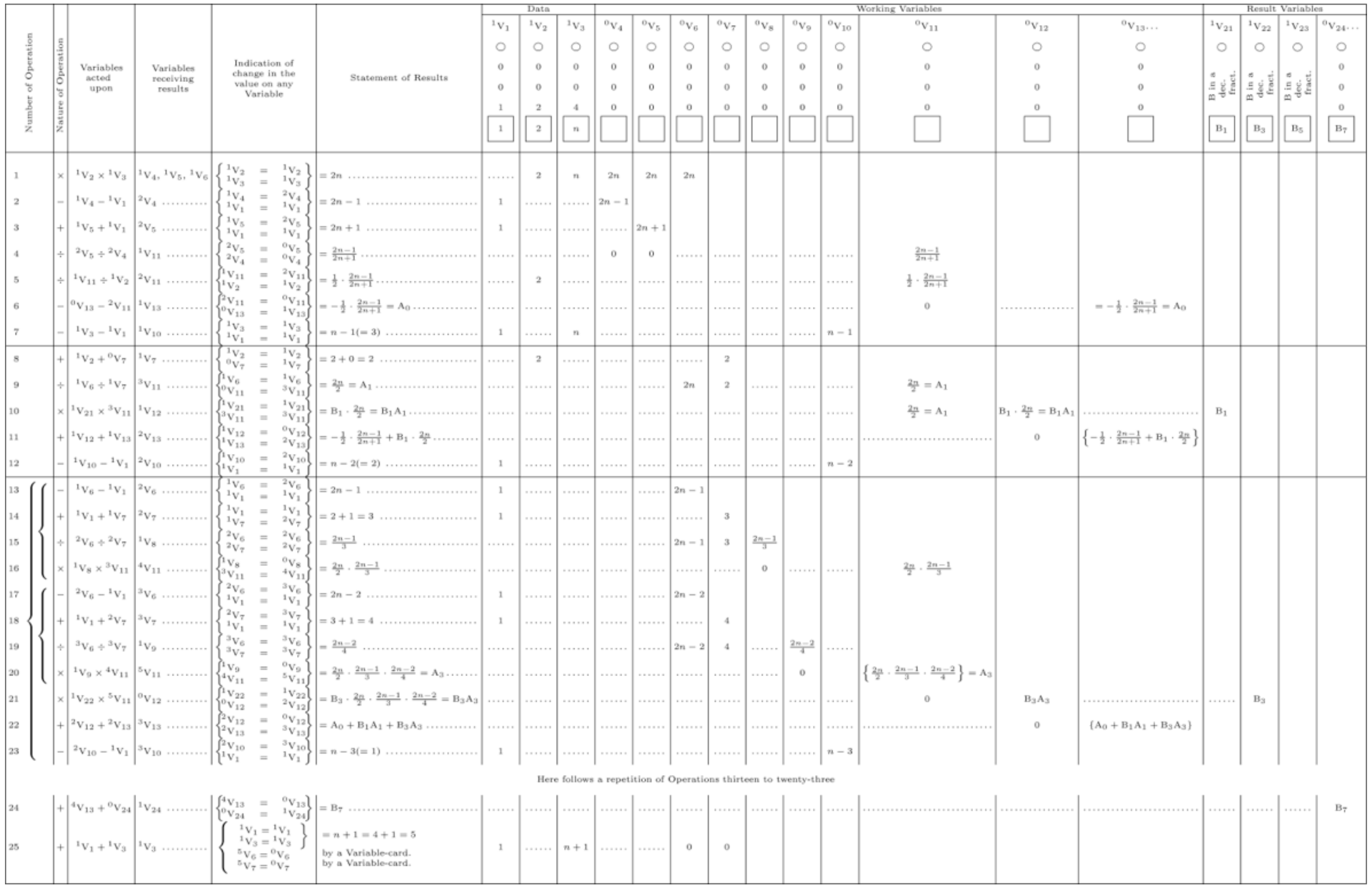
Her ideas included the invention of the program loop and the subroutine!

Ada Byron (Lady Lovelace)
1815-1852

"We may say most aptly that the Analytical Machine weaves algebraic patterns just as the Jacquard loom weaves flowers and leaves." (lovelace)

Friends with Babbage and contributed many ideas for programming the machine

Program for Computing Bernoulli numbers on the analytical engine (1842)



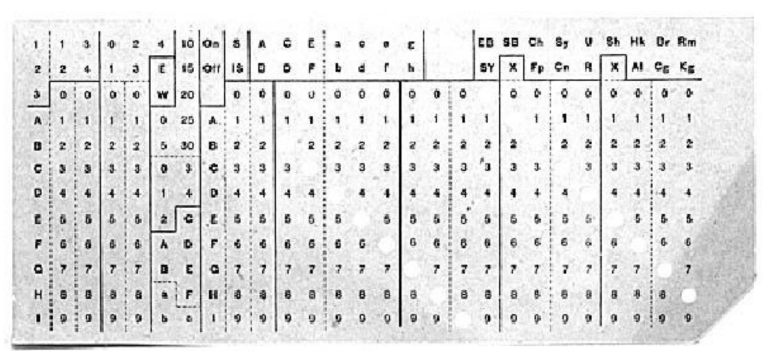
Number of Operation	Nature of Operation	Variables acted upon	Variables receiving results	Indication of change in the value on any Variable	Statement of Results
1	×	$^1V_2 \times {}^1V_3$	$^1V_4, {}^1V_5, {}^1V_6$	$^1V_2 = {}^1V_2$; $^1V_3 = {}^1V_3$	$= 2n$
2	−	$^1V_4 - {}^1V_1$	$^2V_4$	$^1V_4 = {}^2V_4$; $^1V_1 = {}^1V_1$	$= 2n - 1$
3	+	$^1V_5 + {}^1V_1$	$^2V_5$	$^1V_5 = {}^2V_5$; $^1V_1 = {}^1V_1$	$= 2n + 1$
4	÷	$^2V_5 \div {}^2V_4$	$^1V_{11}$	$^2V_5 = {}^0V_5$; $^2V_4 = {}^0V_4$	$= \frac{2n-1}{2n+1}$
5	÷	$^1V_{11} \div {}^1V_2$	$^2V_{11}$	$^1V_{11} = {}^2V_{11}$; $^1V_2 = {}^1V_2$	$= \frac{1}{2} \cdot \frac{2n-1}{2n+1}$
6	−	$^0V_{13} - {}^2V_{11}$	$^1V_{13}$	$^2V_{11} = {}^0V_{11}$; $^0V_{13} = {}^1V_{13}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} = A_0$
7	−	$^1V_3 - {}^1V_1$	$^1V_{10}$	$^1V_3 = {}^1V_3$; $^1V_1 = {}^1V_1$	$= n - 1 (= 3)$
8	+	$^1V_2 + {}^0V_7$	$^1V_7$	$^1V_2 = {}^1V_2$; $^0V_7 = {}^1V_7$	$= 2 + 0 = 2$
9	÷	$^1V_6 \div {}^1V_7$	$^3V_{11}$	$^1V_6 = {}^1V_6$; $^0V_{11} = {}^3V_{11}$	$= \frac{2n}{2} = A_1$
10	×	$^1V_{21} \times {}^3V_{11}$	$^1V_{12}$	$^1V_{21} = {}^1V_{21}$; $^3V_{11} = {}^3V_{11}$	$= B_1 \cdot \frac{2n}{2} = B_1A_1$
11	+	$^1V_{12} + {}^1V_{13}$	$^2V_{13}$	$^1V_{12} = {}^0V_{12}$; $^1V_{13} = {}^2V_{13}$	$= -\frac{1}{2} \cdot \frac{2n-1}{2n+1} + B_1 \cdot \frac{2n}{2}$
12	−	$^1V_{10} - {}^1V_1$	$^2V_{10}$	$^1V_{10} = {}^2V_{10}$; $^1V_1 = {}^1V_1$	$= n - 2 (= 2)$
13	−	$^1V_6 - {}^1V_1$	$^2V_6$	$^1V_6 = {}^2V_6$; $^1V_1 = {}^1V_1$	$= 2n - 1$
14	+	$^1V_1 + {}^1V_7$	$^2V_7$	$^1V_1 = {}^1V_1$; $^1V_7 = {}^2V_7$	$= 2 + 1 = 3$
15	÷	$^2V_6 \div {}^2V_7$	$^1V_8$	$^2V_6 = {}^2V_6$; $^2V_7 = {}^2V_7$	$= \frac{2n-1}{3}$
16	×	$^1V_8 \times {}^3V_{11}$	$^4V_{11}$	$^1V_8 = {}^0V_8$; $^3V_{11} = {}^4V_{11}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3}$
17	−	$^2V_6 - {}^1V_1$	$^3V_6$	$^2V_6 = {}^3V_6$; $^1V_1 = {}^1V_1$	$= 2n - 2$
18	+	$^1V_1 + {}^2V_7$	$^3V_7$	$^2V_7 = {}^3V_7$; $^1V_1 = {}^1V_1$	$= 3 + 1 = 4$
19	÷	$^3V_6 \div {}^3V_7$	$^1V_9$	$^3V_6 = {}^3V_6$; $^3V_7 = {}^3V_7$	$= \frac{2n-2}{4}$
20	×	$^1V_9 \times {}^4V_{11}$	$^5V_{11}$	$^1V_9 = {}^0V_9$; $^4V_{11} = {}^5V_{11}$	$= \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = A_3$
21	×	$^1V_{22} \times {}^5V_{11}$	$^0V_{12}$	$^1V_{22} = {}^1V_{22}$; $^0V_{12} = {}^2V_{12}$	$= B_3 \cdot \frac{2n}{2} \cdot \frac{2n-1}{3} \cdot \frac{2n-2}{4} = B_3A_3$
22	+	$^2V_{12} + {}^2V_{13}$	$^3V_{13}$	$^2V_{12} = {}^0V_{12}$; $^2V_{13} = {}^3V_{13}$	$= A_0 + B_1A_1 + B_3A_3$
23	−	$^2V_{10} - {}^1V_1$	$^3V_{10}$	$^2V_{10} = {}^3V_{10}$; $^1V_1 = {}^1V_1$	$= n - 3 (= 1)$
				Here follows a repetition of Operations thirteen to twenty-three	
24	+	$^4V_{13} + {}^0V_{24}$	$^1V_{24}$	$^4V_{13} = {}^0V_{13}$; $^0V_{24} = {}^1V_{24}$	$= B_7$
25	+	$^1V_1 + {}^1V_3$	$^1V_3$	$^1V_1 = {}^1V_1$; $^1V_3 = {}^1V_3$; $^5V_6 = {}^0V_6$; $^5V_7 = {}^0V_7$	$= n + 1 = 4 + 1 = 5$; by a Variable-card. by a Variable-card.

The programming language ADA is named in her honor

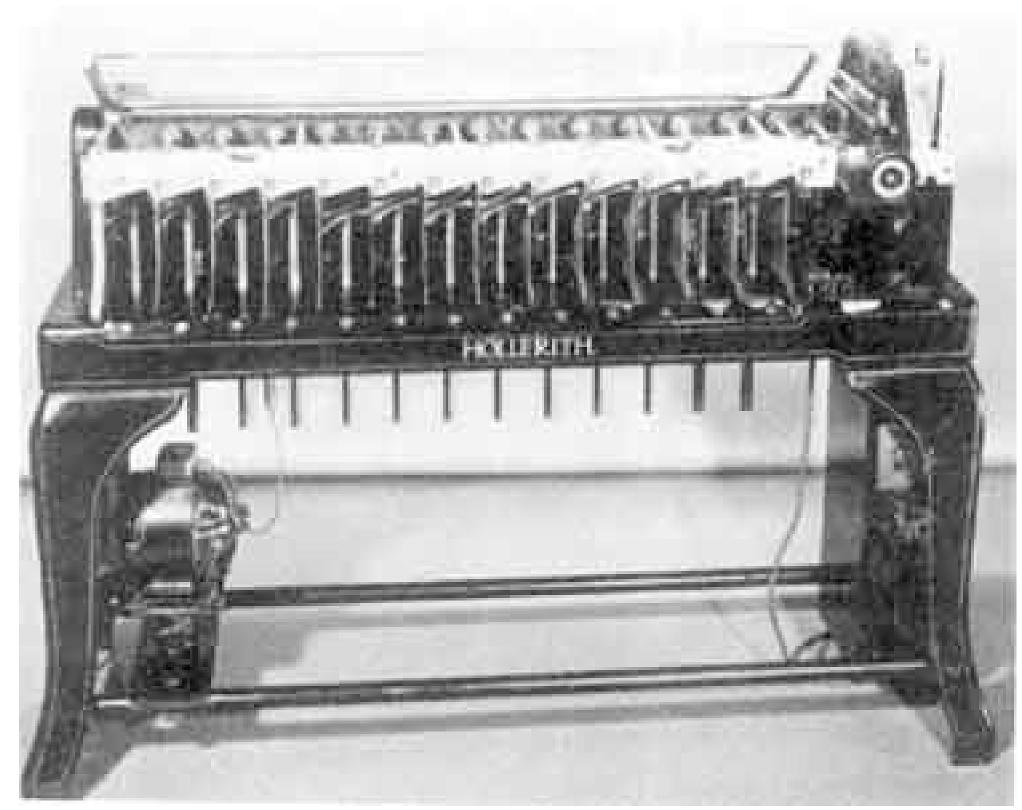
Herman Hollerith

- Developed a calculating method for the 1890 census
- Based on Hollerith code using cards with holes punched in them
- 1896 founded the Tabulating Machine Company
- 1911 merged with another company to form the Computing Tabulating Company (CTR) with Thomas J. Watson as President.
- 1924 the company was renamed **IBM**!

Commercial data processing!



Hollerith Punch Card



Hollerith Electric Tabulating System

Electronics Roots of Computation

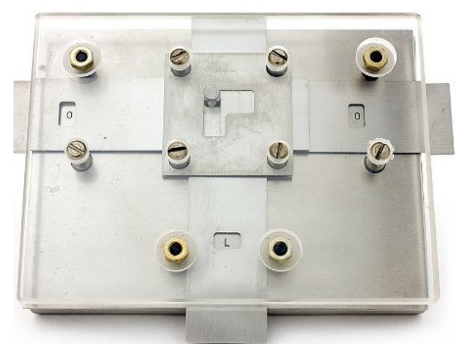
[Electronic Advances]

Electro-mechanical relay
Electro-magnetic relay
Vacuum Tube
Transistor
Large Scale Integration
Very Large Scale Integration

Konrad Zuse: Z1

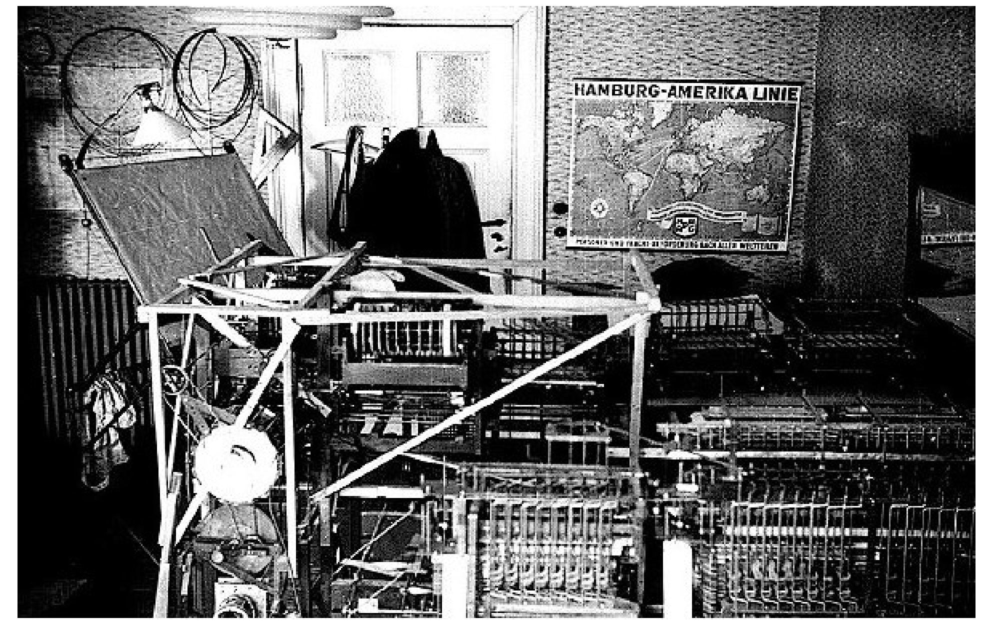
First binary computer/calculator

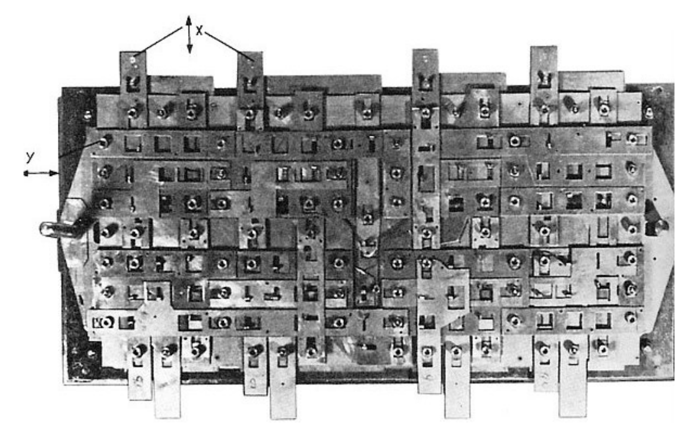
Konrad Zuse
1910-1995



Mechanical binary gate

Z1 - Mechanical Calculator
built from an Erector Set! (1938)



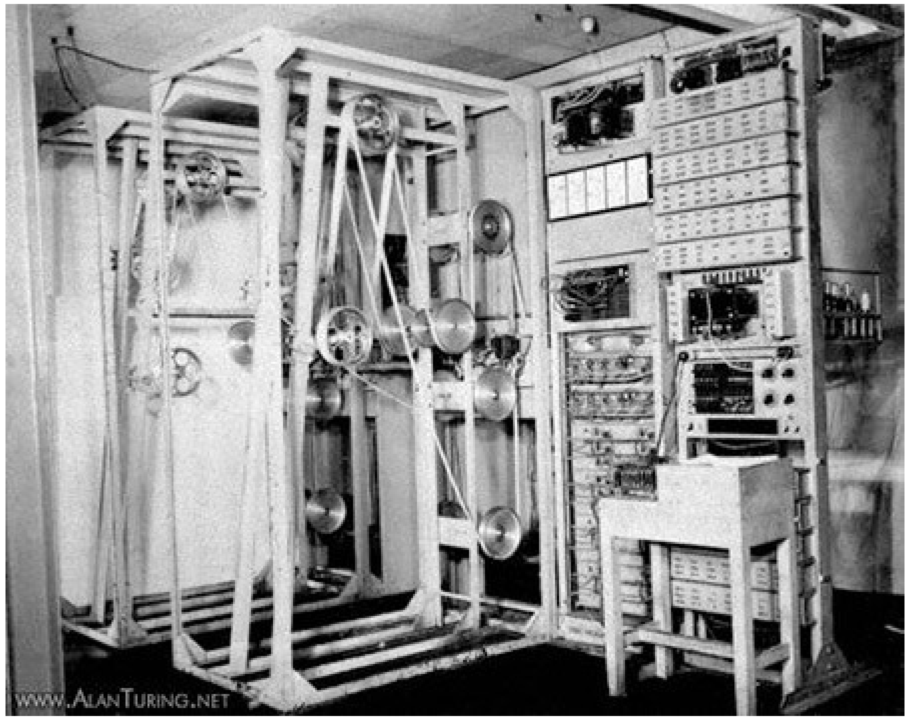


Mechanical Circuits

<u>Z2</u> (1939) was an improvement on the Z1, using <u>electro-mechanical telephone relays</u>.
It was not programmable but could solve simultaneous equations.
First fully functioning electro-mechanical computer.

Heath Robinson Machine

- World War 2:
  Two major German ciphers
  - Enigma – machine with 3 wheels
  - Lorentz – machine with 12 wheels, more advanced
- **<u>Heath Robinson</u>**: Hardware that helped decipher the Lorentz cipher
  - 1st <u>operational</u> special purpose computer (early 1940s)
  - Electro-mechanical Relays
  - May have shortened the war by 2 years



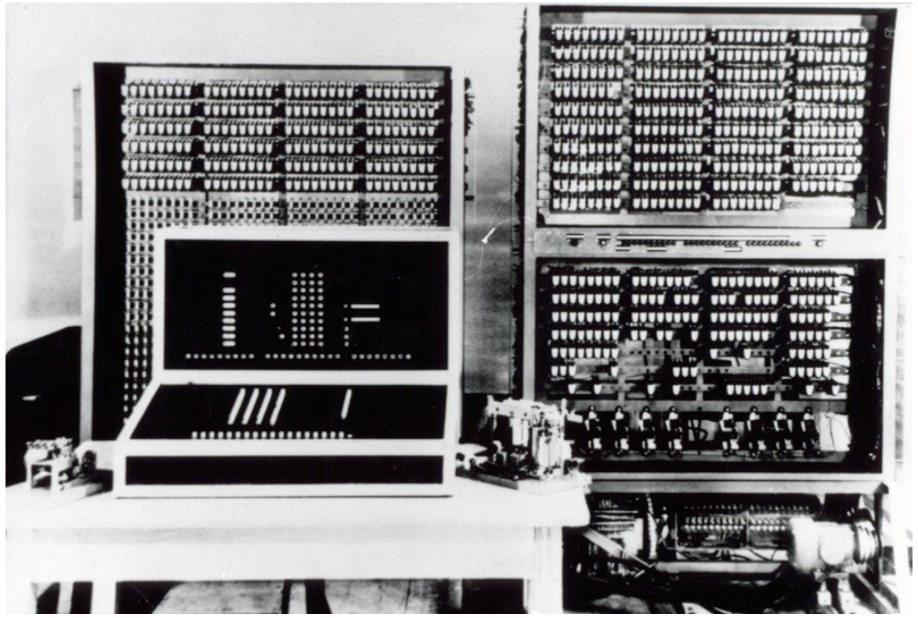
Konrad Zuse: Z3

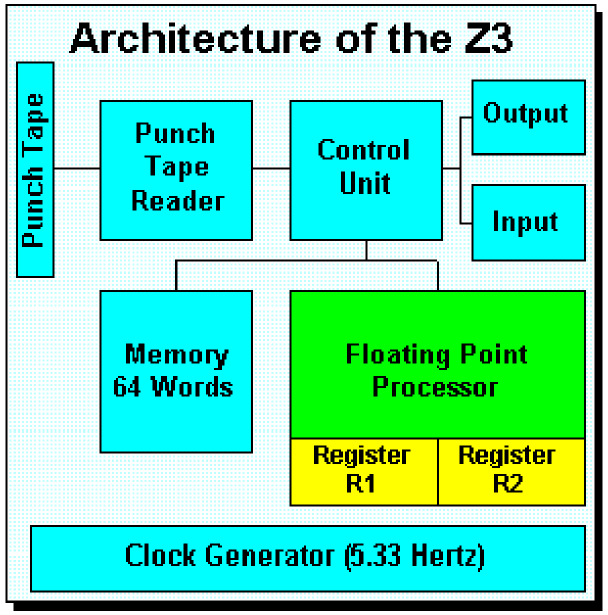
Z3 was the world's first programmable digital computer (late 1941).

- Memory of 64 words, 22 bits each (1408 bits)
- 1408 electro-mechanical relays to support the random access memory
- Another 1200 relays for the central processing unit.
- No stored programs

Programming the Z3

- Had loops without conditional jumps. Included arrays and records.
- Supported floating point operations
- Zuse developed the Plankalkul (1945), the world's first high-level programming language (no goto with a structured programming methodology)

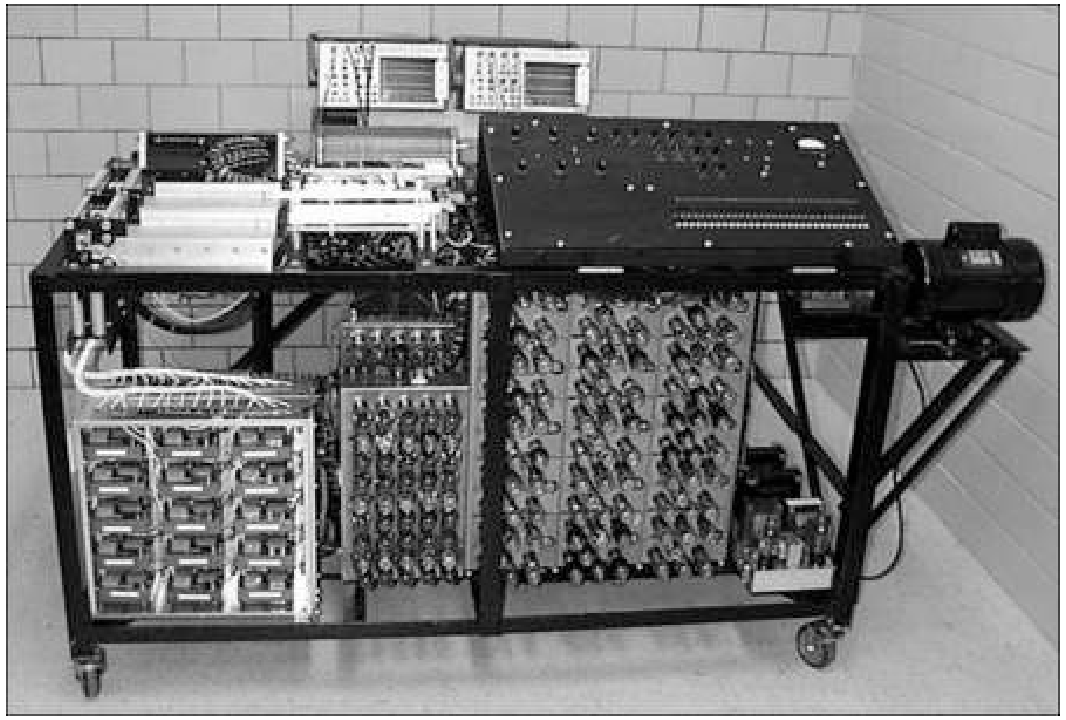


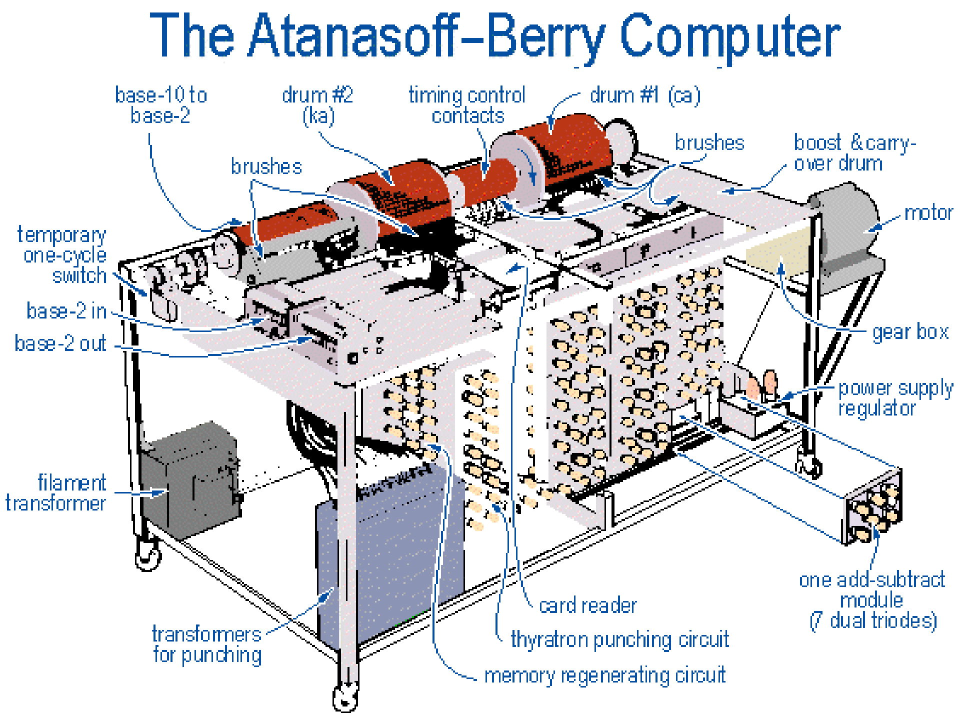


LINKÖPING UNIVERSITY

Atanasoff-Berry: ABC [1940]

The Atanasoff-Berry Computer (1939-42) was the first (special purpose) electronic digital computer built from Vacuum Tubes and Capacitors





700 pounds, over 300 vacuum tubes, a mile of wire.
One operation every 15 seconds (as opposed to billions per second).

Not reprogrammable, but the first to implement three critical ideas:

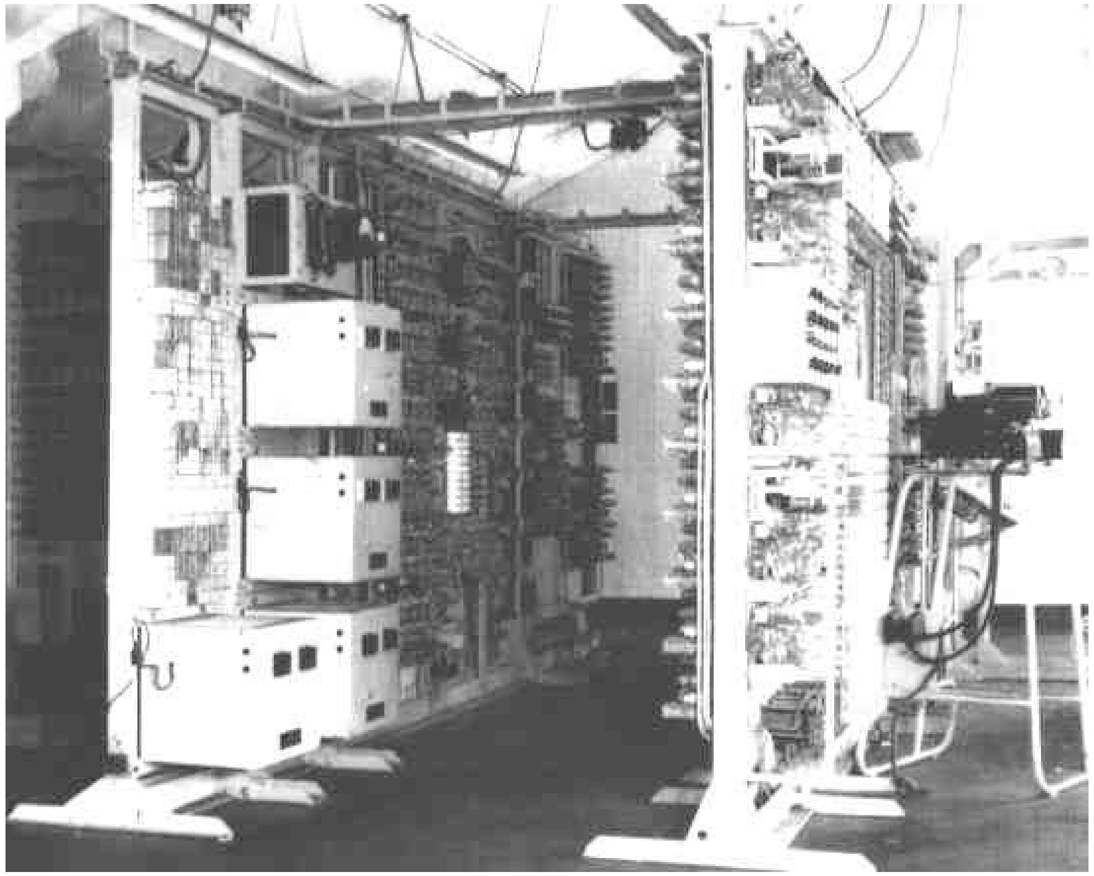
1. Using binary digits to represent all numbers and data
2. Performing all calculations using electronics rather than wheels, ratchets, or mechanical switches
3. Organizing a system in which computation and memory are separated.

The system pioneered the use of regenerative capacitor memory, as in the DRAM still widely used today.

Colossus Machine

First English Electronic Computer (1943)

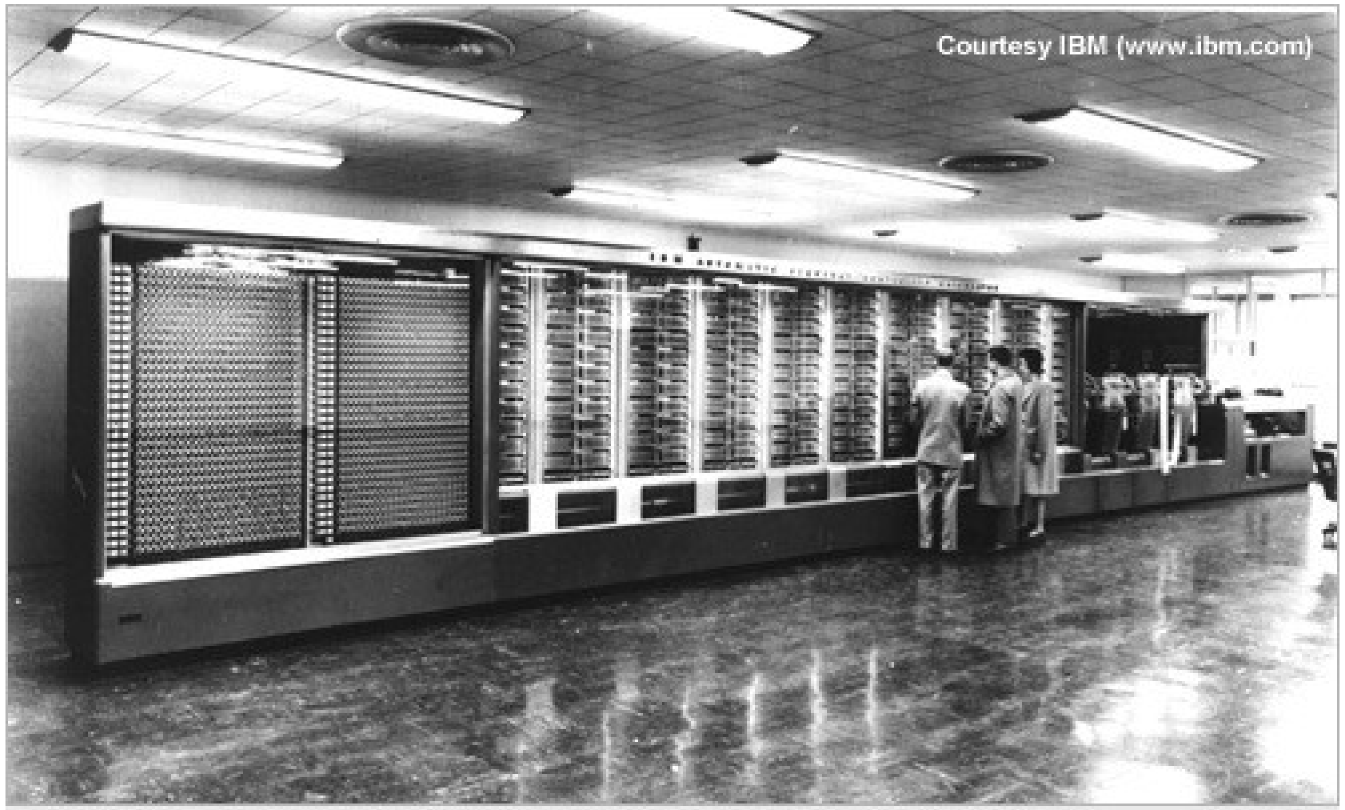
- 1500 Tubes, not programmable. Specialized for code breaking.
- Reduced the time to break messages from weeks to hours
- By the end of the war 63 million characters of high grade German messages had been intercepted.



Mark I Computer (1944)

First American General Purpose Reprogrammable Computer

- Developed by Harvard and IBM scientists led by Howard Aiken
- Much influenced by Babbage's Analytical Machine
- Formal Name: **IBM** Automatic Sequence Controlled Calculator!
- Electro-Mechanical: switches, relays, rotating shafts..



The Harvard Mark I

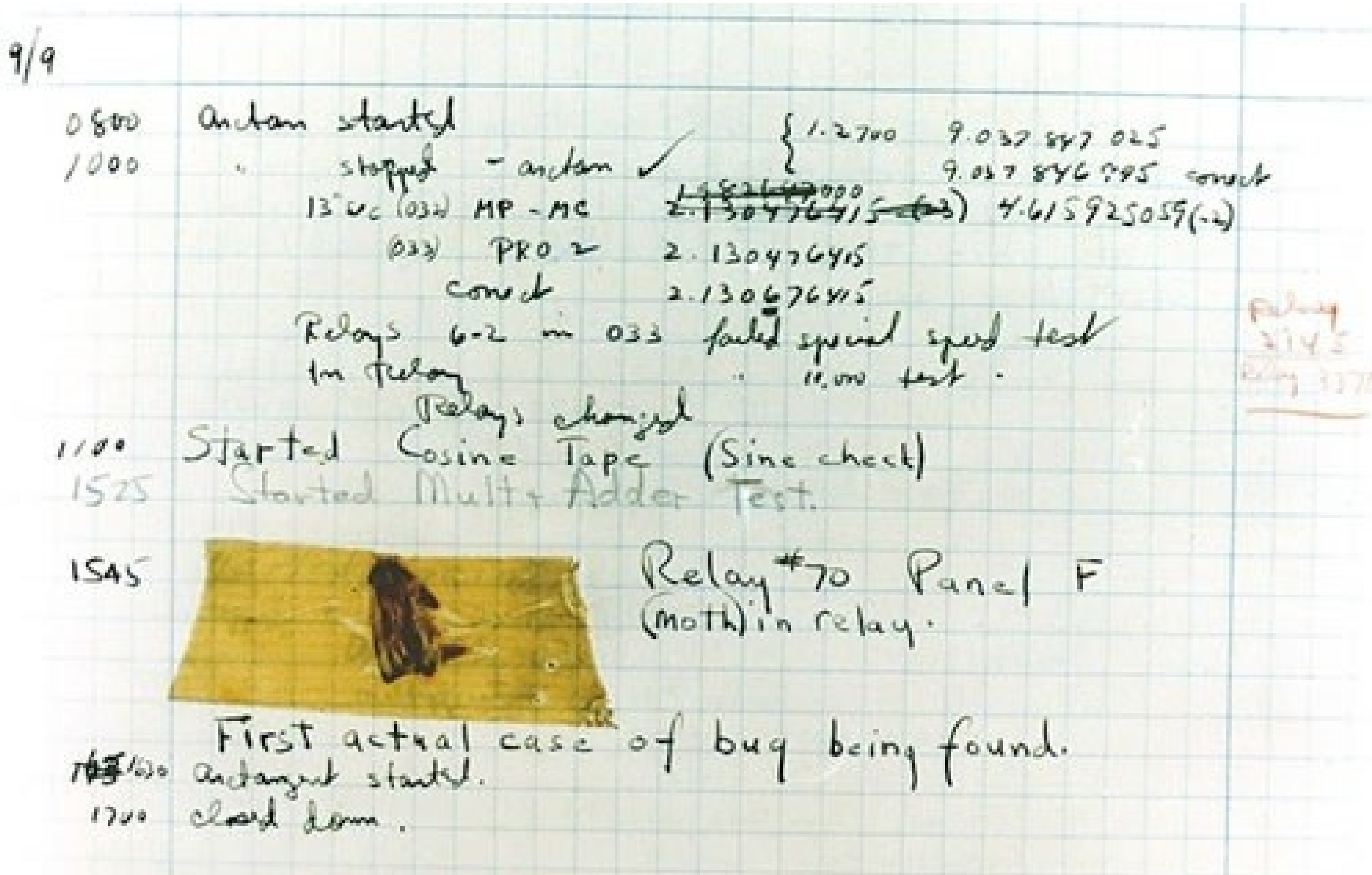
IBM Archives:

The Automatic Sequence Controlled Calculator (Harvard Mark I) was the first operating machine that could execute long computations automatically. A project conceived by Harvard University's Dr. Howard Aiken, the Mark I was built by IBM engineers in Endicott, N.Y. A steel frame 51 feet (16 m) long and eight feet high held the calculator, which consisted of an interlocking panel of small gears, counters, switches and control circuits, all only a few inches in depth. The ASCC used 500 miles **(800 km) of wire** with three million connections, 3,500 multipole relays with 35,000 contacts, 2,225 counters, 1,464 tenpole switches and tiers of 72 adding machines, each with 23 significant numbers. It was the industry's largest electromechanical calculator

Rear Admiral Grace Murray Hopper

- Considered the Ada Lovelace of the Mark I
- Credited with having written the first high-level language compiler
- Led the effort to develop COBOL (Common Business Oriented Language)

The first computer "Bug".



Mark I Computer's Log Book

After a moth was removed from a relay in the Mark I to fix a problem:
"From then on, whenever anything went wrong with the computer, we said we had bugs in it. If anyone asked if we were accomplishing anything, we replied we were "debugging".

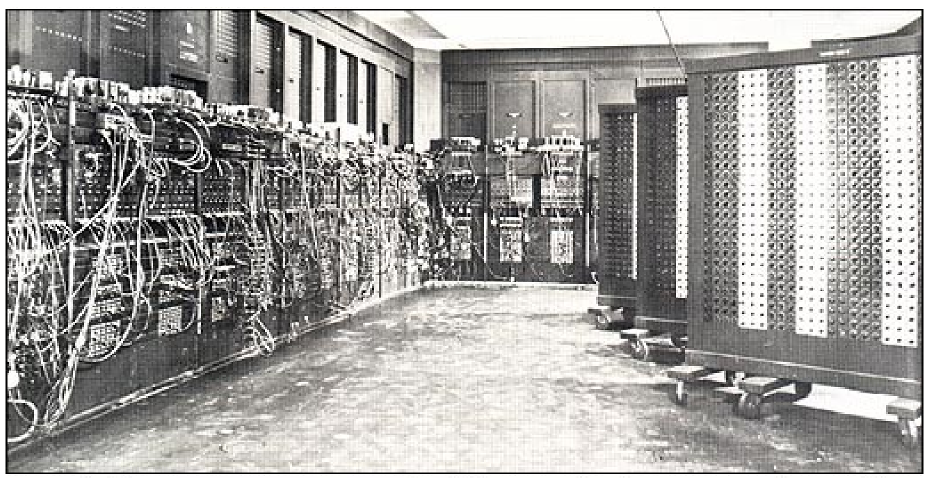
LINKÖPING UNIVERSITY

ENIAC [1946]

The world's first fully electronic general purpose computer!

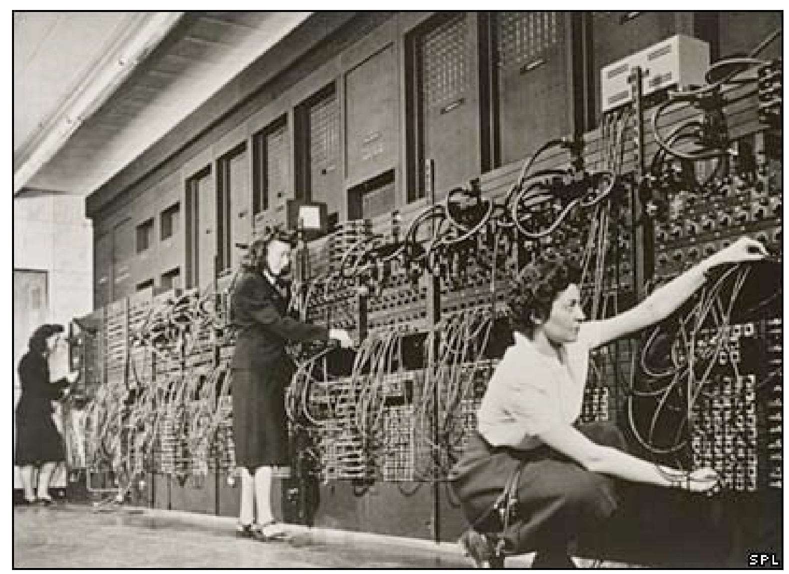
<u>ENIAC - Electronic Numerical Integrator and Computer</u>

- Developed by J. Presper Eckert, Jr. and John W. Mauchly at University of Pennsylvania
- **18,000 vacuum tubes, 5000 additions per second** (1000 times faster than the Mark I)
- Major problem:
  - Loading a program involved setting 6000 switches and connecting hundreds of cables!
  - A new concept was waiting to be (re-)discovered: **<u>Stored programs</u>**.

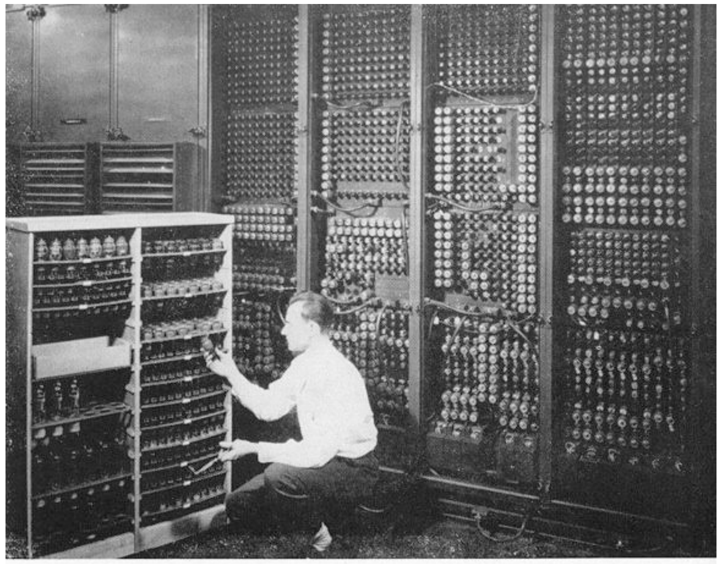


Among the first assignments given to Eniac, first all-electronics digital computer, was a knotty problem in nuclear physics. It produced the answer in two hours. One hundred engineers using conventional methods would have needed a year to solve the problem

The ENIAC



Reprogramming



Replacing a bad tube meant checking among ENIAC's 19,000 possibilities.

Replacing a Tube

The ENIAC contained 17,468 vacuum tubes, along with 70,000 resistors, 10,000 capacitors, 1,500 relays, 6,000 manual switches and 5 million soldered joints. It covered 1800 square feet (167 square meters) of floor space, weighed 30 tons, consumed 160 kilowatts of electrical power. 1000 bit memory.

In one second, the ENIAC (one thousand times faster than any other calculating machine to date) could perform 5,000 additions, 357 multiplications or 38 divisions.

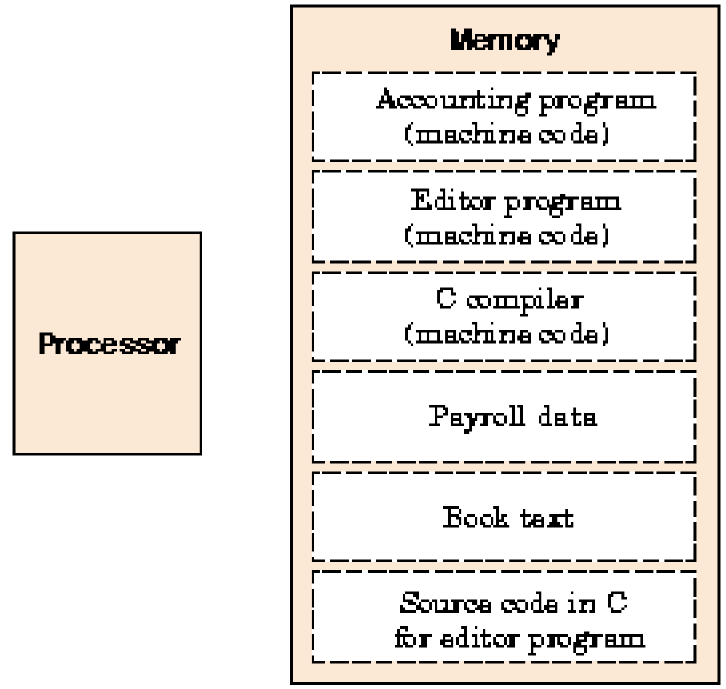
The Stored Program Concept

Store a Program as Data in Memory

Von-Neumann wrote a memo about the ENIAC formalizing its ideas!

Precursors: Babbage,Turing…

John von Neumann
1903 - 1957



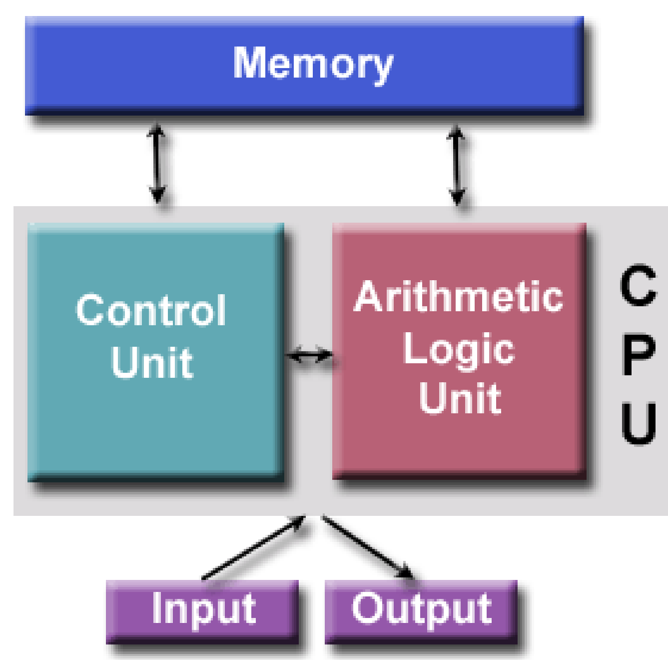
Program

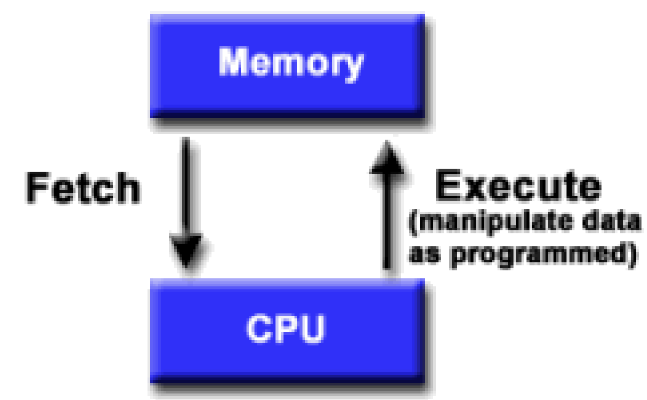
Data

Memory

CPU

I/O





Processor fetches “data” from memory and interprets it as program instructions

Supports:

- Self modifying code
- subroutines
- recursion

LINKÖPING UNIVERSITY

Stored Program Computers

- EDVAC (1949)
  - World's 1st Stored Program Computer conceived
  - Construction delayed due to Eckert and Mauchly starting a company, but finished in 1951
- EDSAC (early 1949)
  - 1st Stored program computer built
  - Wilkes ...(England)
- BINAC (late 1949)
  - 1st American stored program computer built
  - Eckert and Mauchly
  - Eckert and Mauchly's company sold to Remington-Rand

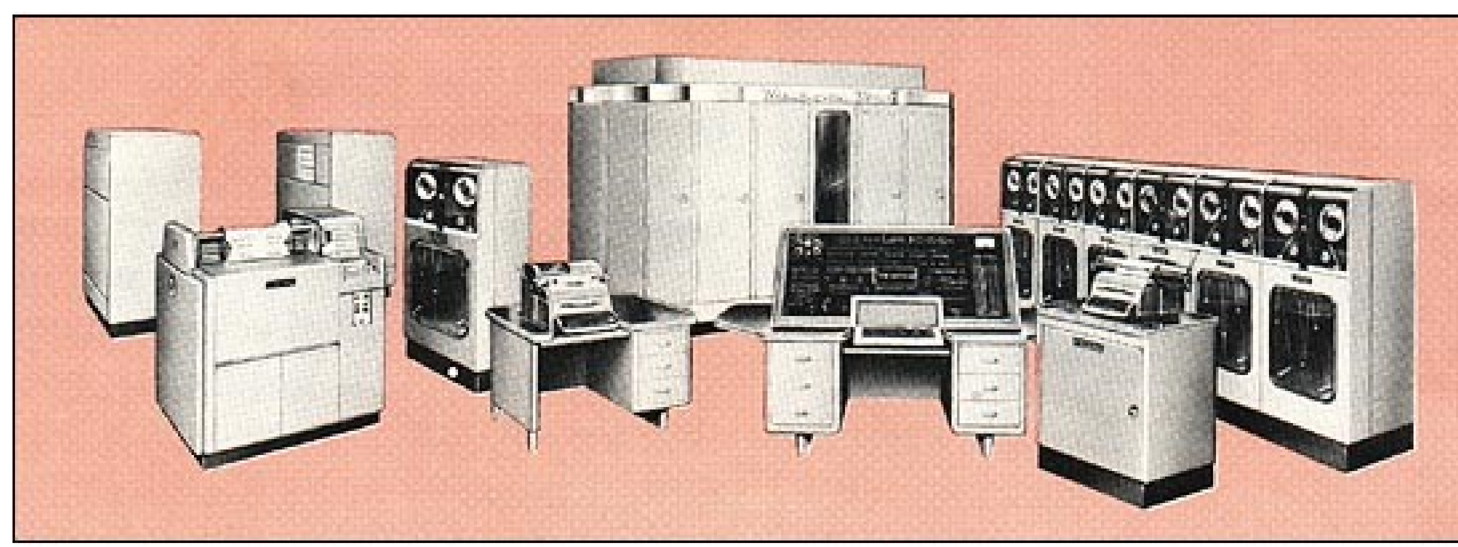
UNIVAC I [1951]

- World's 1st commercially marketed electronic computer
  - **UNIV**ersal **A**utomatic **C**omputer
    - Inventors: Eckert and Mauchly
- Built by Remington-Rand
  - Originally priced at $159,000 but rose to between $1,250,000 - $1,500,000.
  - Sold 46 in total
  - Rumored Market Study: potential world-wide market for 50 computers!
  - Not a commercial success
- Specification:
  - 5200 vacuum tubes, 13 metric tons, 125kW
  - 1905 instructions per second
  - Main memory: 1000 words of 12 characters
  - External memory: magnetic tapes
  - 1st customer: U.S. Census bureau

UNIVAC I

Inside the UNIVAC

The control console (HMI!)



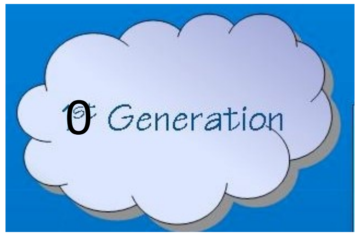
Univac 1 computer, the first business data processor. Shown, left to right, High-speed printer with power supply and magnetic tape input unit, Unityper II, the central computer, the control console, the Universo 1 input-output unit, and the Uniprinter

IBM 700 Series

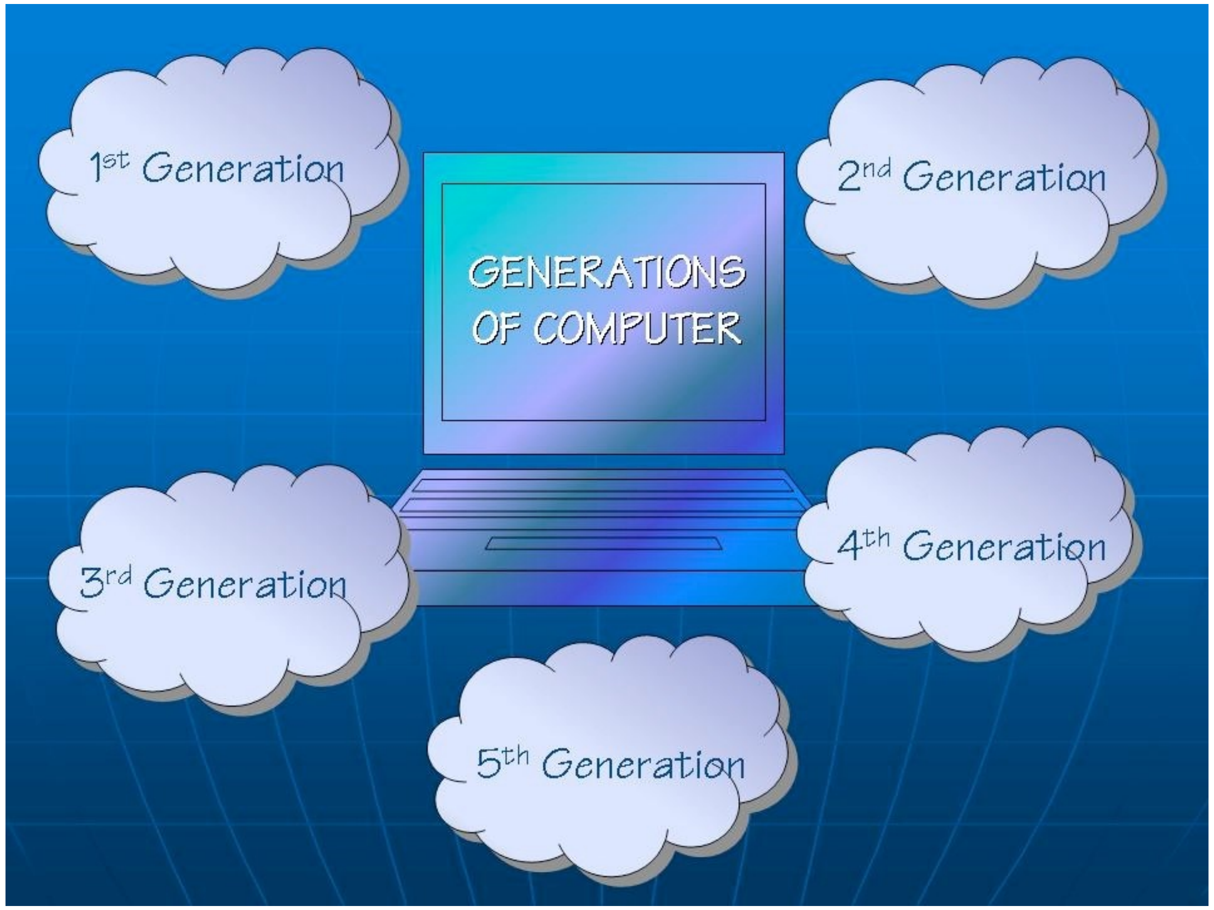
- 1st commercially successful general purpose computers
  - 1952 Nathaniel Rochester
  - IBM 701 Electronic Data Processing Machine
- Large scale (main frame) computers
  - Used vacuum tube logic
  - 36 bit/18 bit numbers
  - Main memory: 4096 - 36 bit binary words with 6 bit characters
- IBM positioned itself to become world dominant
- Led to the development of the programming language FORTRAN

Computer Generations

Computer Generations provide a coarse division of progress in computer technology due to a technological innovation that fundamentally changes the way computers operate.

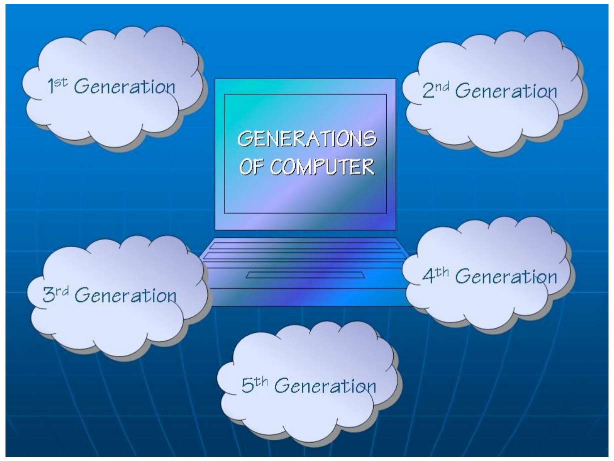


- 0th Generation (?? - ??)
- Mechanical Calculators
- Abacus 500 BC
- Pascaline 1642 (Pascal)
- Step Reckoner 1671 (Leibniz)
- No memory
- Human intervention at all levels

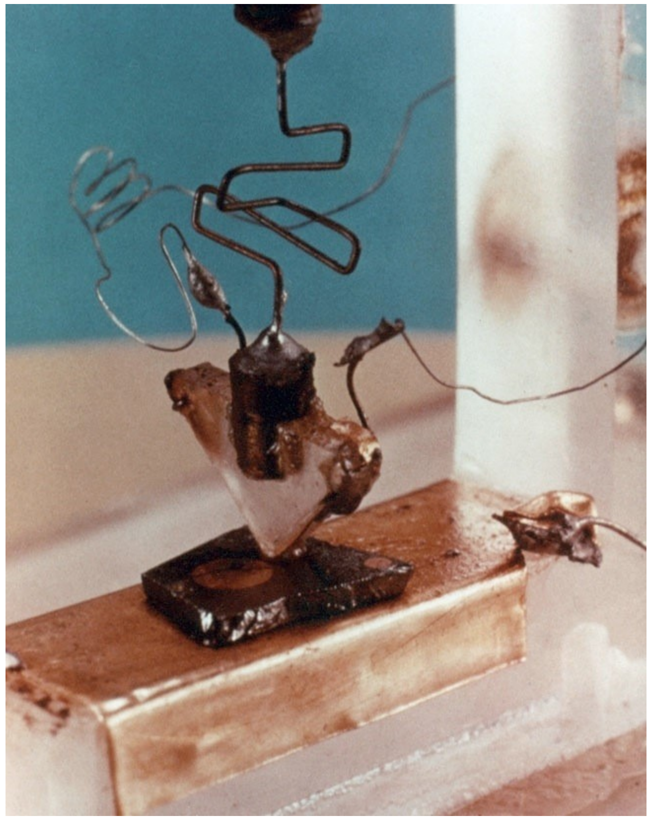


1st Generation (1940 - 1956)

Hardware	Vacuum tubes for Circuitry	
Memory	Magnetic drums for secondary memory	No immediate access memory
Programming	Machine language	
High-Level Language	None	
Input	Punch cards and paper tape	
Output	Printouts	



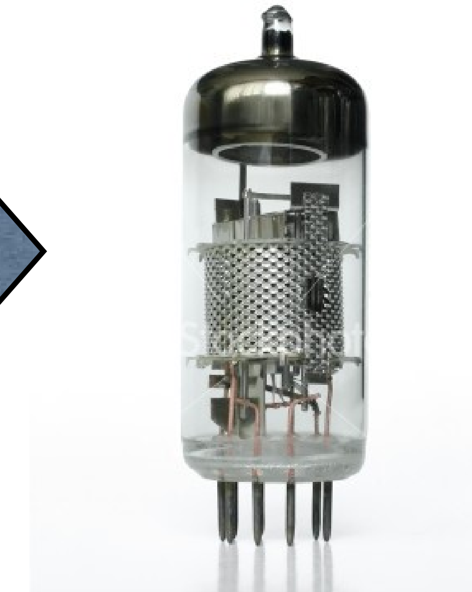
2nd Generation (1956-1963)



1st Transistor

Transistors

Replaces



Vacuum tube

The name transistor comes from the '*trans*' of transmitter and '*sistor*' of resistor

Transistor is a semiconductor device used to amplify and switch electronic signals and electrical power

- In 1948, Bardeen, Brattain, Shockley invented the transistor at Bell labs
- A solid state version of the vacuum tube that uses germanium and silicon, which are semi-conductors
- Could act as both a transmitter of electricity and a resistor, controlling electronic current.
- Lower power consumption, smaller, more reliable, cheaper and much lower heat dissipation

2nd Generation Computers

- Generation 2 computers were still bulky and expensive. Basically used by big business, government agencies and universities.
- Big computer vendor companies were formed:
  - **IBM**
    - IBM7094: used for scientific applications (1962)
    - IBM 1401: used for business applications (1959)
  - **DEC** PDP 1(1960)
  - **CDC** 6600: “first” supercomputer (1964)
    - $10 million
    - 10 million instructions/sec, 60 bit words, 128k word of memory
    - World’s fastest computer from 1964-69
    - Built by a team led by Seymour Cray
- Transistor-based computers enabled space travel and many other advances.

PDP I [1960]

- Developed by **DEC** (Digital Equipment Corporation)
  - Most important computer in the creation of hacker culture at MIT, BBN, …
  - First commercial computer that focused on interaction with the human user rather than the efficient use of computer cycles
  - Had an optional high-resolution graphical display that MIT students used to play Spacewar!, the first interactive computer video game.
  - Specs:
    - 18 bit words and 4 kilowords for main memory (9 kbytes)
    - clock speed around 200 KHz
    - punched paper tape as primary storage medium

The PDP-1's operating system was the first to allow multiple users to share the computer simultaneously. This was perfect for playing Spacewar, which was a two-player game involving warring spaceships firing photon torpedoes. Each player could maneuver a spaceship and score by firing missiles at his opponent while avoiding the gravitational pull of the sun.

Multi-User System

Graphical Display

IBM 7094 [1962]

Large IBM 7094 System

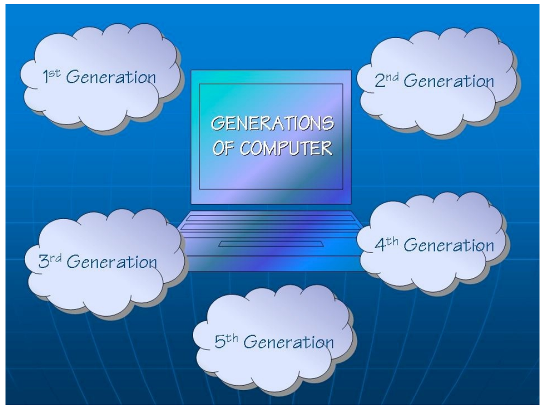
1962:
7000 series replaced 700 series

Advent of large Data Centres

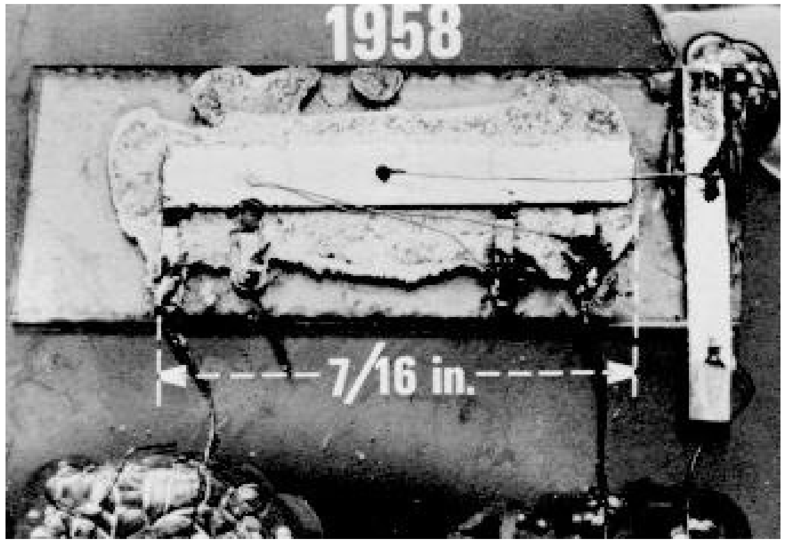
NASA Computer Room controlled Mercury & Gemini space flights

2nd Generation [1956 - 1963]

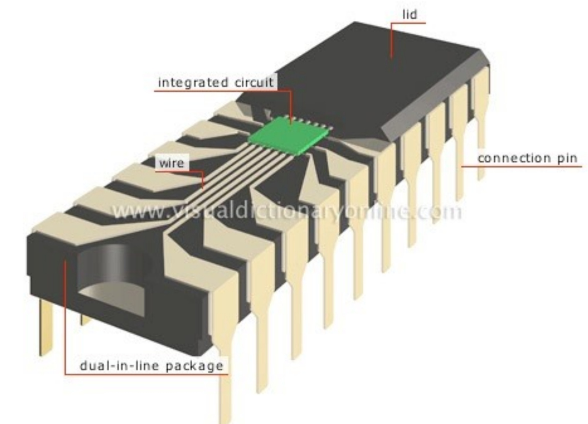
Hardware	Transistors for Circuitry	
Memory	Magnetic disks for secondary memory	Advent of immediate access memory
Programming	From machine language to assembly language	
High-Level Language	Algol, COBOL FORTRAN	
Input	Punch cards and paper tape (some displays) (	
Output	Printouts (some displays)	



3rd Generation [1963 - 1971]



1st Integrated Circuit



Packaged Integrated Circuit

- In the late 1950’s Kilby and Noyce independently came up with the idea of an **integrated circuit** (IC)
  - Single miniature circuit with dozens of components (transistors, resistors, …)
    instead of individual components to be connected

IBM System 360

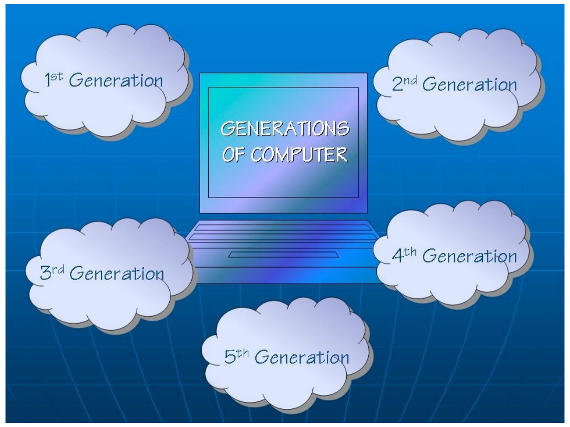
The first computers to be built entirely with IC's

- System/360 series replaced the 7000 series
  - Small to large, low to high performance all using the same instruction set
  - System/360 announced in 1964, available in 1965
  - OS360
- New innovations
  - Introduced the 8 bit byte
  - IBM floating point architecture
  - EBCDIC character set
  - Nine-track magnetic tapes
- Continues to form the basis for IBM mainframes even today...

IBM System 360

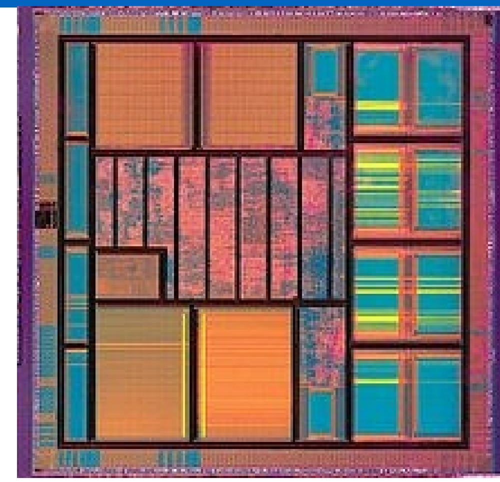
3rd Generation [1963-1971]

Hardware	Integrated circuits miniaturised transistors	
Memory	Magnetic drums for secondary memory	Immediate access memory
Programming	Assembly languages High-level languages	Operating systems becoming mainstream
High-Level Language	Basic, Interlisp, Pascal	
Input	Keyboards	
Output	Monitors	

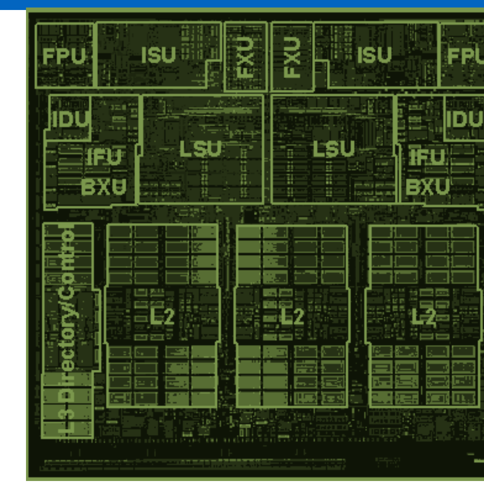


4th Generation [1971 -1984]

The advent of the Microprocessor



VLSI Die

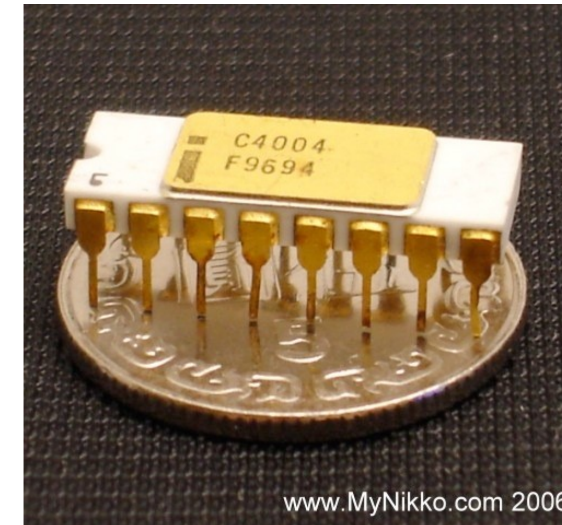


Typical VLSI Chip

- More and more transistors on a **single chip**…
  - SSI (small scale integration): 10-100
  - MSI (medium scale integration): 100-1000
  - LSI (large scale integration): 1000-10000
  - VLSI (very large scale integration): >10000
    - Possible to have a full CPU on a single chip – a microprocessor
    - 2008: billion-transistor processors are available
    - Apple M1 Ultra: 114 billion transistors (and flash memory, far more regular, had >2000 billion even in 2019)

Microprocessors

- 4004 microprocessor (World’s first single chip MP!)
    - Created by Intel in 1971
    - 2250 transistors (Apple A15 has ~15 billion, including GPU)
    - 4-bit processor, 8 bit wide instruction set, speeds up to 740kHz,
    - Could address 4K of ROM and 1280x4 bits of RAM



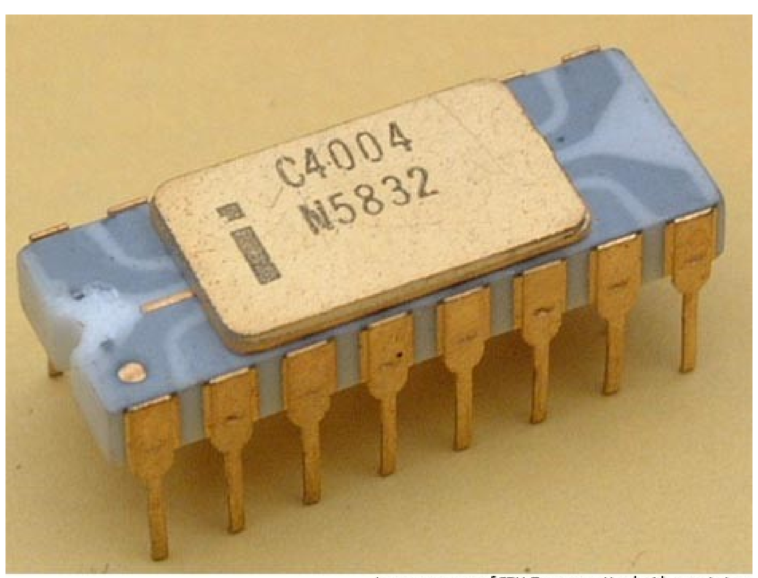
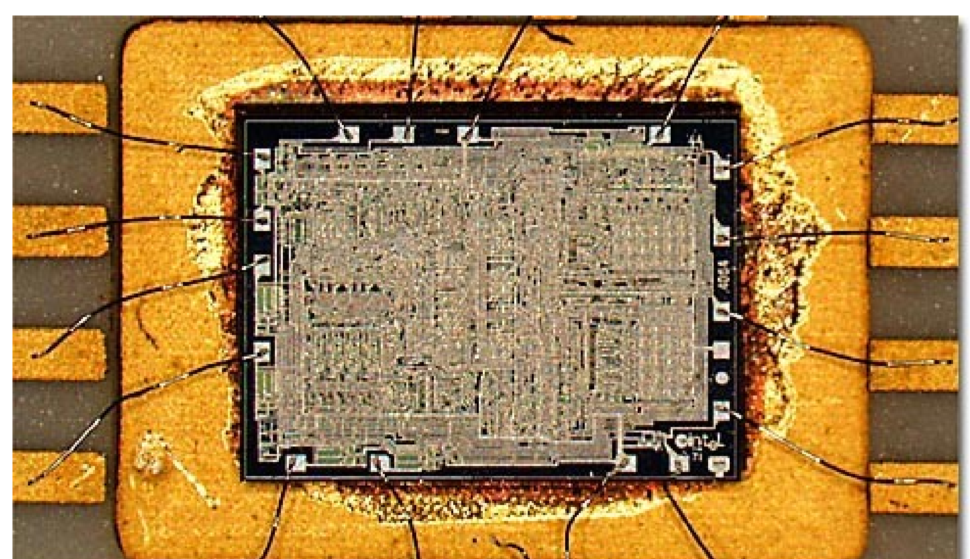
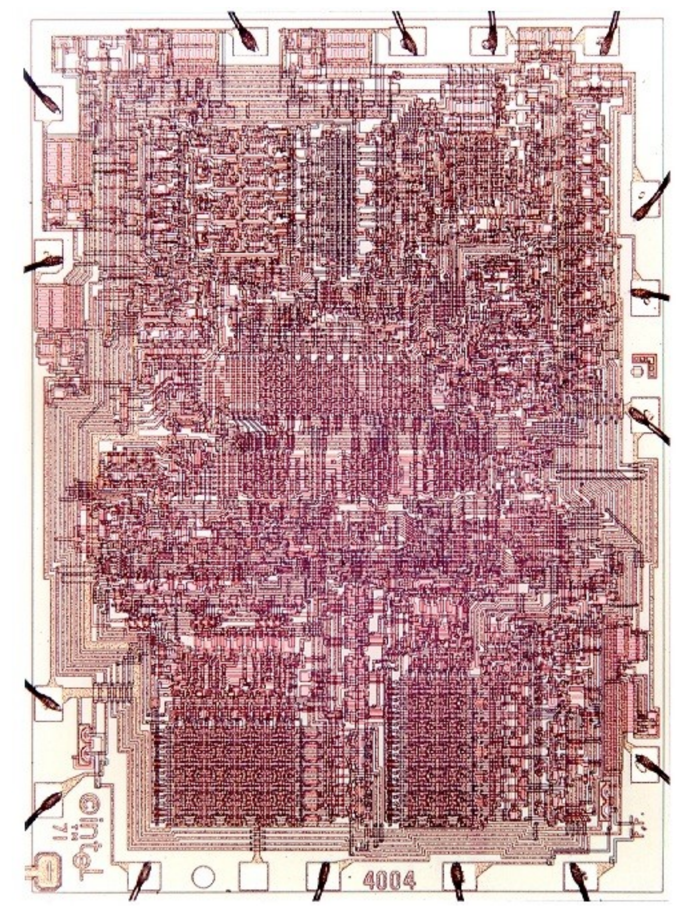


Image courtesy of CPU-Zone.com. Used with permission.

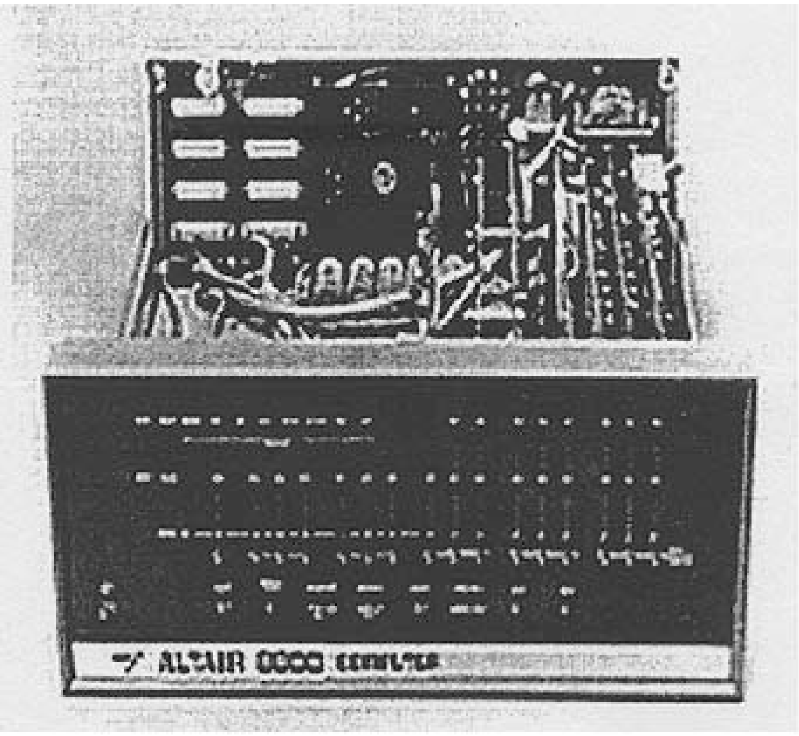


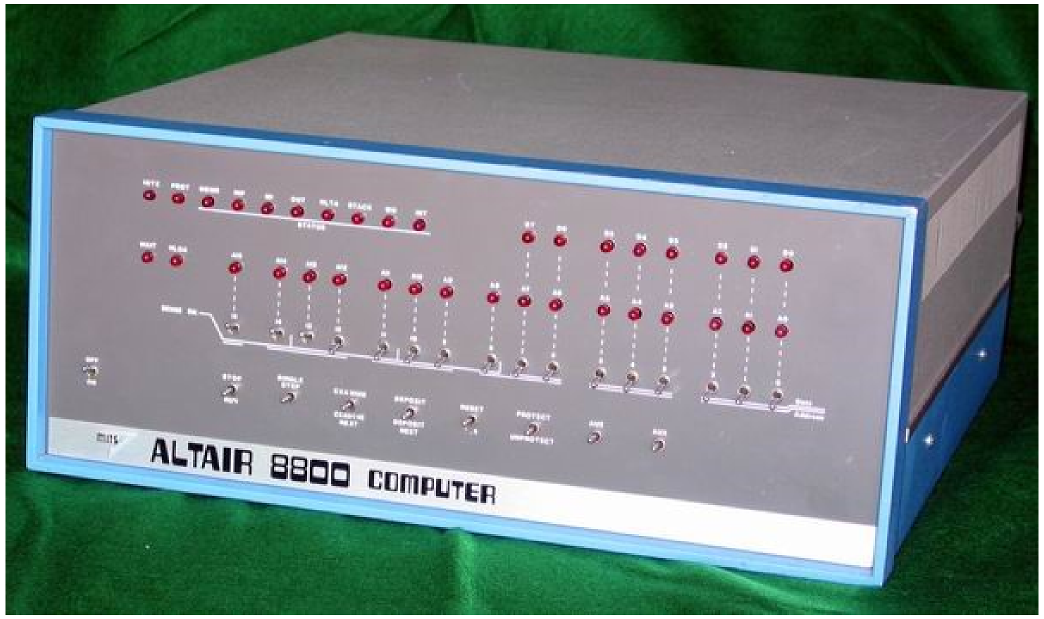


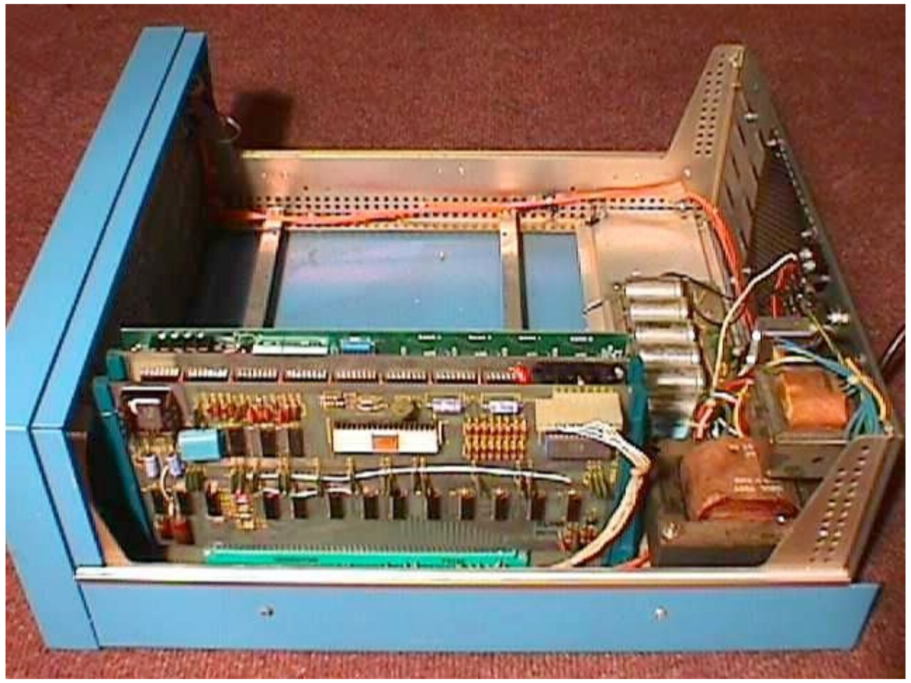
Personal Computers [1975]

World's 1st Personal Computer

- MITS Altair (1975)
  - 2 MHz Intel 8080 chips
  - 256 byte memory (not kbytes!!!)
  - Was just a box with flashing lights
  - $395 for a kit, $495 assembled.



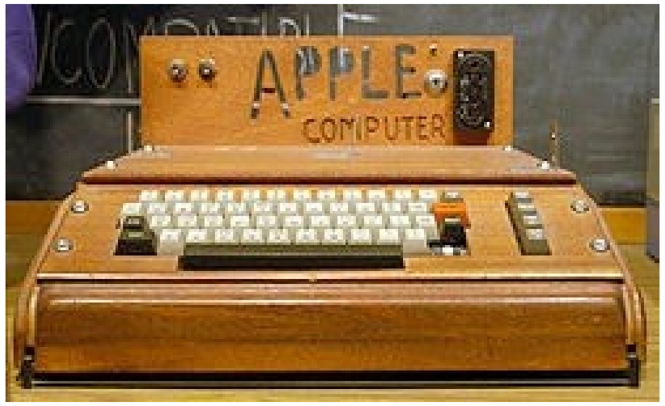




Apple Computers [1976]

- Founded in 1976 by Steve Jobs, Steve Wozniak and Ronald Wayne
  - Purpose: to sell the Apple 1 personal computer kit
  - Hand Built by Wozniak and 1st shown to the public at the Homebrew Computer Club
  - Specs:
  - Sold as a motherboard (with CPU, RAM and basic textual-video chips)
  - $666.66 in 1976

Released: July 1976 (discontinued September 1977).
Price: $666.66
Memory: 4K RAM (expandable to 8KB or 48KB using expansion cards).
How many? About 200 total.
CPU: MOS 6502, 1.0 MHz.
Display: Monochrome 280 X 192, 40 X 24 text.
Keyboard: Not included.
Ports: Composite video output keyboard interface, one vertical expansion slot.
Storage: Cassette interface available.
Song storage capacity: Zero



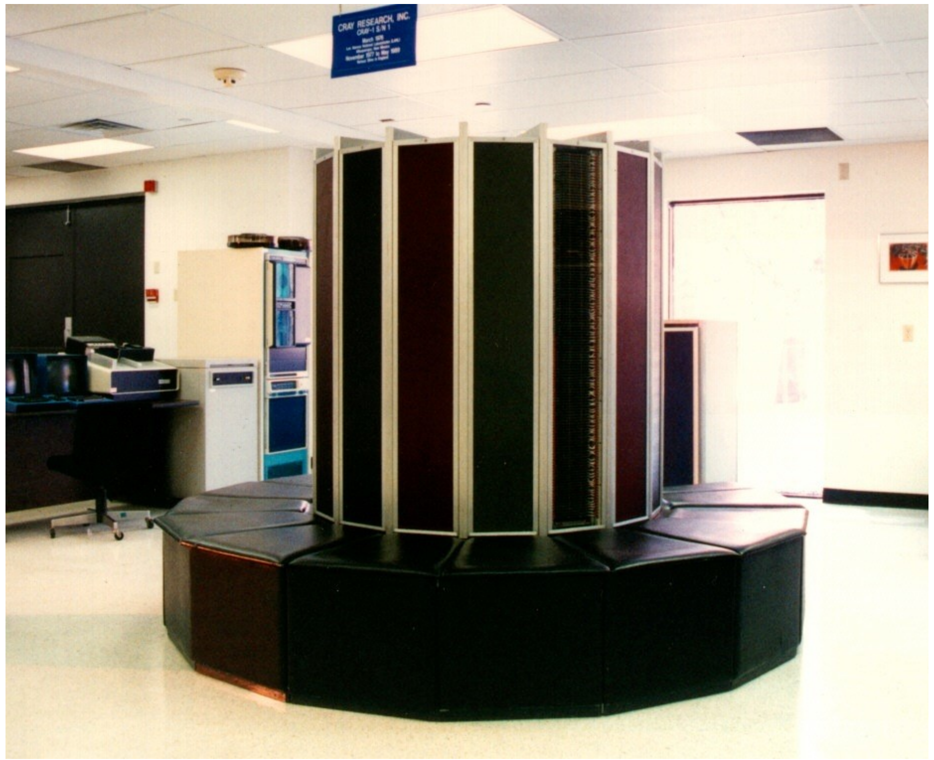
Customized Wood Enclosure

The Kit!

Jobs/Wozniak

CRAY I [1976]

- Seymour Cray created the Cray Research Corporation (1972)
  - Father of super-computing
- Cray-I (1976) -- $8.8 million
  - 160 million instructions per second, 8Mbytes of memory
  - Used Freon for refrigeration, no wire in the system longer than 4 feet. Circular design to keep the integrated circuits close together
  - 1st one installed at Los Alamos National Laboratory in 1976



LiU!

APPLE II [1977]

- Apple II (1977)
  - Character cell based color graphics
  - Open Architecture
  - 5 1/4 inch floppy disk drive and interface
  - Desktop computer
    - ***Killer App***: VisiCalc (spreadsheet)

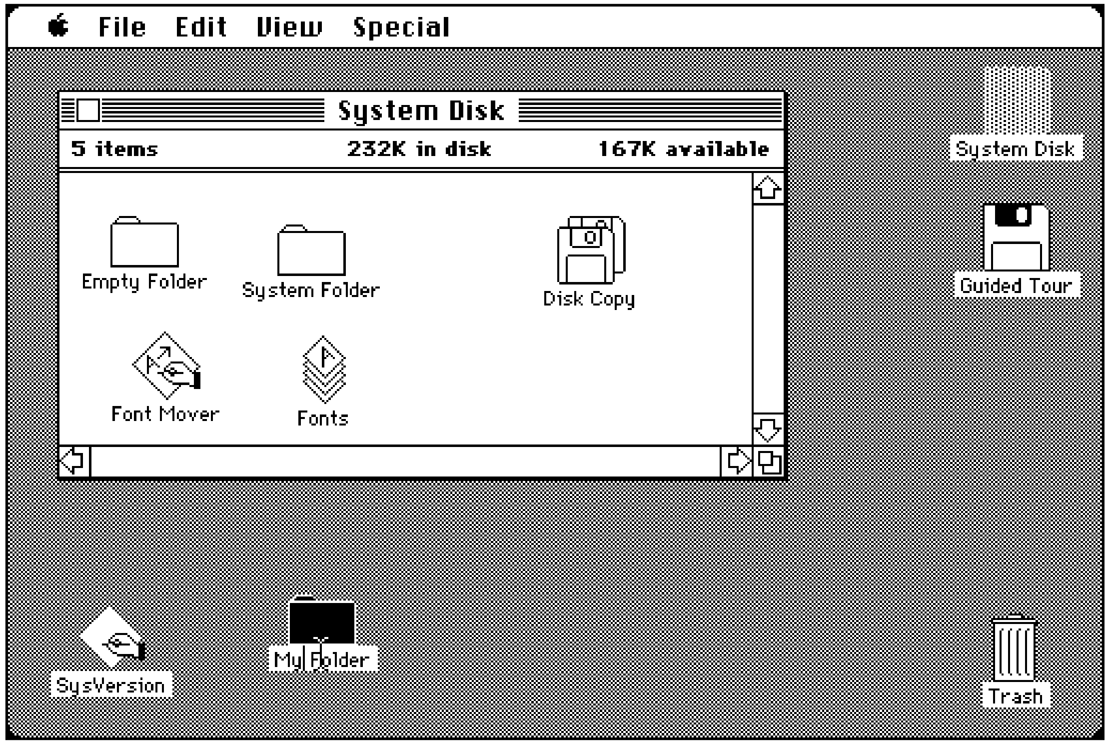
IBM PC 5150 [1981]

- Introduced in 1981, developed by a team of about 12 people in about a year
  - Used an Intel 8080 processor
  - Bill Gates was approached in 1980 to talk about writing a new operative system for IBM's Personal Computer
    - MS-DOS
  - Pricing started at $1,565 and up..
- IBM-PC became a standard due to its success

Macintosh [1984]

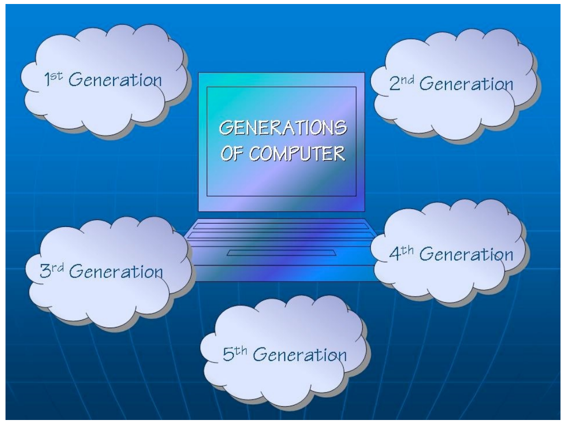
- **Macintosh** (1984)
  - Advanced graphics capabilities (one of the 1st GUIs)
    - Windows, Icons, Mouse
    - 1st commercially successful computer to use a GUI
  - LaserWriter introduced: 1st Postscript laser printer
  - PageMaker introduced: desktop publishing package.
- Created the desktop publishing market!

LISA



4th Generation [1971 - 1984]

Hardware	Very large-scale integrated circuits Micro-Processors	Personal Computers, Workstations
Memory	Semi-conductor memories	
Programming	High-level languages	Unix, Networking
High-Level Language	Smalltalk, C, Prolog, Pascal	
Input	Keyboards	
Output	Monitors	



5th Generation [?? - ??]

- Advent of Artificial Intelligence (AI)
- Advanced audio visual interfaces (HCI)
- Multi-Core processors
- Parallel processing and superconductors
- Quantum Computation
- Nano-Technology
- Molecular Computing
- Ubiquitous Computing
- Cloud Computing
- The Internet of Things
- Smart Phones
- The singularity!

Ray Kurzweil

The technological **singularity** (also, simply, the **singularity**) is the hypothesis that the invention of artificial superintelligence will abruptly trigger runaway technological growth, resulting in unfathomable and unforeseeable changes to human civilization.

The present and beyond!