History & Ethics

History and Ethics of Computer Science

CS51 – Fall 2025

Welcome!

Our team

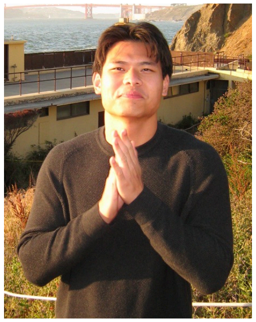
Miriam Brody
she/her/hers

Claudio Castillo
they/them/theirs

Alyssa Coleman
she/her/hers

Ayşegül (Aisha) Kula
she/her/hers

Victoria (Vika) Prokopenko
she/her/hers



Connor Wang
he/him/his

Who are you?

- Preferred name and pronouns
- Programming experience
  - Concurrently enrolled in CSCI050, or
  - Placed out of CSCI050 after talking with Prof. Chen

Nice to meet you!

What is CS51?

- An introduction to computer science through a survey of fundamental topics and a glimpse into our curriculum.
  - Week 1: History and Ethics of Computer Science
  - Weeks 2-5: Computer Systems (preview of CSCI105)
  - Weeks 6-9: Mathematical Foundations of Computer Science (preview of CSCI054)
  - Weeks 9-11: Data Structures and Algorithms (preview of CSCI062 and CSCI140)
  - Weeks 12-13: Theory of Computation and Programming Languages (preview of CS101)
  - Weeks 14-15: Machine Learning (preview of electives)

How can I succeed in CS51?

- Sleep well the night before, eat, come to class, be on time (I know, it’s early)
- Take notes, participate, ask questions, don’t stay confused
- Review slides and do the assigned problems after each lecture
- Start the assignments **early**
- Come to office hours/mentor sessions
- Budget at least 8 hours outside class time.
- Read email/Slack for announcements and bookmark course website:
    - https://cs.pomona.edu/classes/cs51/

How can I be a good citizen in CS51?

- Use laptops/tablets/phones/other fancy electronics only for note taking/live coding.
- Be mindful when in office hours/mentor sessions of other students waiting for help.
  - Come with specific questions after you tried the problem!
- TAs are students, too. Respect their time outside mentor sessions.
- We encourage collaboration but we want you to submit your own assignments.
- We monitor assignments for plagiarism. This includes using code from other students, websites, or tools like coPilot or chatGPT.
  - Academic honesty violations are reported to the Dean of Students. Assignments will receive a zero. Exams will receive a zero and half a grade is reduced. Second infraction leads to failure of the course.
  - If unsure about what's allowed, talk to me.

Average week

- Tuesday/Thursday lectures
- Friday labs
- 10 weekly assignments, generally due on Thursday midnight

Grading summary

- Weekly Assignments: 35%
  - Four free days - can use on one assignment or across different assignments.
  - If group assignment, both partners have to use a free day.
  - Let me know **before** the deadline if you will take a late day pass.
- Midterm I: 15% (September 26^{th}, in lab)
- Midterm II: 15% (November 7^{th}, in lab)
- Final Exam: 30% (December 12^{th}, 2-5pm, in this classroom)
  - If traveling during winter break, please book your tickets accordingly to avoid conflicts.
- Lab Attendance: 5%
  - You have to attend lab in person to receive credit. Can skip one lab, no questions asked.

Slack channels

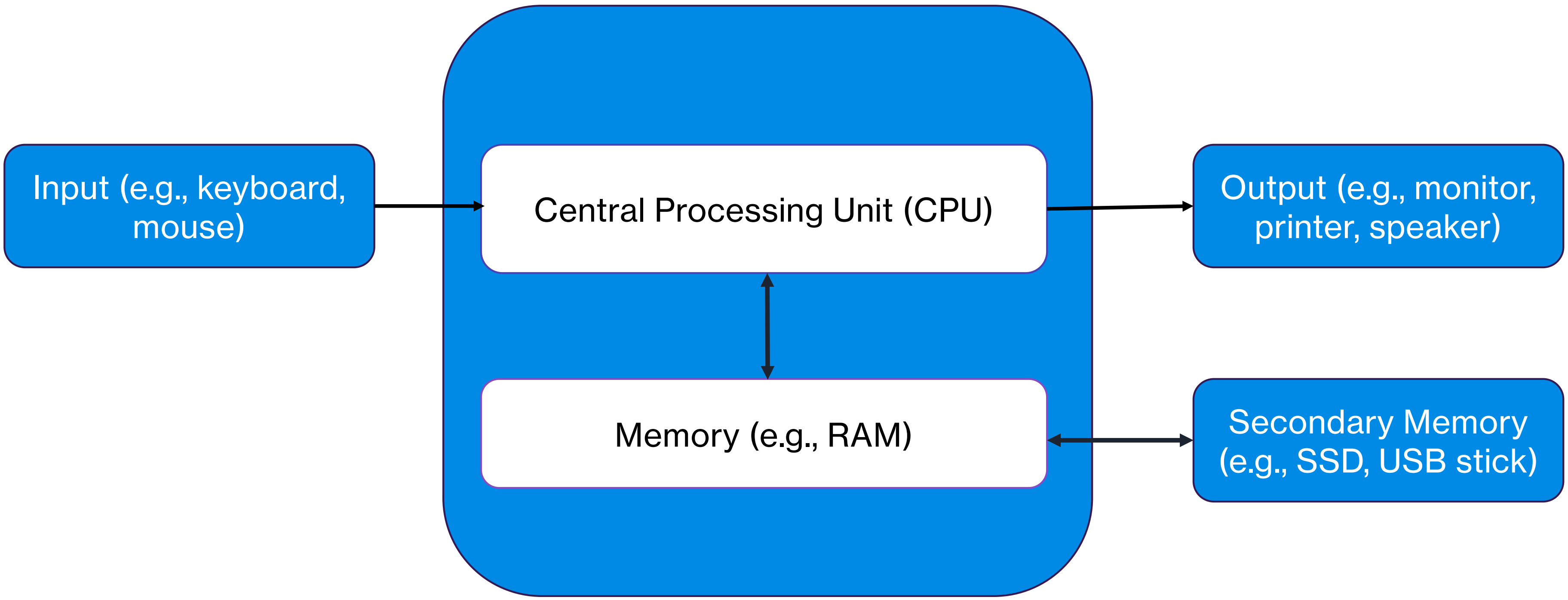
- If registered, you have been invited to cs51-fall2025 channel in http://slack.pomona.edu
- Department-wide Slack workspace: https://tinyurl.com/PomonaCSSlack

Computer Science History

What is a computer?

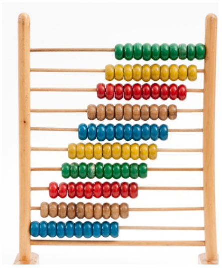
- A **programmable** electronic device that can process, store, and retrieve data.
- A computer processes data according to a set of instructions or a **program**.
- **Hardware**: physical parts of the computer (e.g., CPU, RAM, motherboard, graphics card).
  - **Peripherals**: auxiliary devices that can provide **input** (e.g., mouse, keyboard, microphone, webcam, game controller), receive **output** (e.g., monitor, printer, speaker), provide **storage** (e.g., external drives), and facilitate transmission of data across **networks** (e.g., routers and modems).
- **Software:** programs that instruct the computer what to do.
  - **Operating system (OS)**: intermediary programs managing resources between hardware and applications (e.g., Windows/macOS/Linux for desktops).
  - **Application software**: programs that perform specific tasks for users. (e.g., word processor, media player, Web browser etc.).

A simplified view of a computer

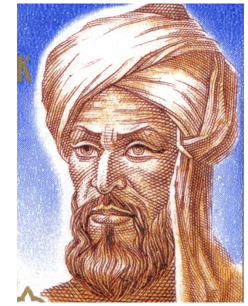


In the beginning was the calculation

- Before computer science was an academic discipline, key ideas in computing existed in the form of **calculation**.
- Early societies had a limited vocabulary for counting, often employing the fingers of one hand, of two hands, or counting all fingers and toes to count in **bases** of 5, 10, or 20, respectively.
  - Western societies today use the decimal system (base 10).
  - But languages like English and French still carry remnants of the base 20 system, For example, the Gettysburg Address starts: "Four *score* (i.e. twenty) and seven years ago”
- The next numeric system that arose about 5000 years ago was base 60 (sexagecimal) and passed from ancient Sumerians to Babylonians in Mesopotamia.
  - Base 60 is used today in the subdivision of 1 hour into 60 minutes, and of 1 minute into 60 seconds.
- Babylonians did basic calculations for agricultural purposes on the **abacus**.



Numeral systems


Al-Khwarizmi

- In the four millennia that followed, tremendous progress was done in mathematics.
- **Numeral systems** allowed to encode numbers and other symbols using consistent mathematical notation.
- By 9th century, Islamic mathematicians extended arithmetic contributions made by Indian mathematicians and popularized the **Hindu-Arabic numeral system** we use.
  - We owe the origin of words like algebra and algorithm to the Persian mathematician **Al-Khwarizmi**.

Computing devices throughout the millennia

- Humans also continued creating computing devices, such as **analog machines** built to do specific calculations or **automata** built to accomplish certain tasks to assist in commerce, navigation, military, and science.
- Each improvement made computations and tasks faster and easier than it was previously possible, amplifying our mental abilities.
- This is a common theme within computer science.
    - In the 19th century, early computer pioneer Charles Babbage famously said, *"At each **increase of knowledge**, as well as of the contrivance of a **new tool**, human **labor becomes abridged**".*

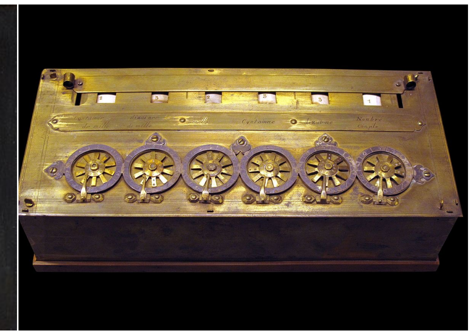
When computers were people

- None of these computing devices were called computers.
- The earliest documented use of the word “computer” is from 1613 in a book by English poet Richard Braithwait, where he describes the profession of a person as
  *"the truest **computer** of all times, and the **best arithmetician** that ever breathed and he reduced thy dayes into a short number."*
- In those days, **computer was a person who carried calculations swiftly and accurately**, sometimes with machines but often not.
- The job title persisted till mid 20th century when it shifted to describe devices instead of people.
- Most computers were women because they could be paid less than their male counterparts.
- This trend continued with the first programmers who were often Black women since they could be paid less than their white counterparts.

Cogs and wheels

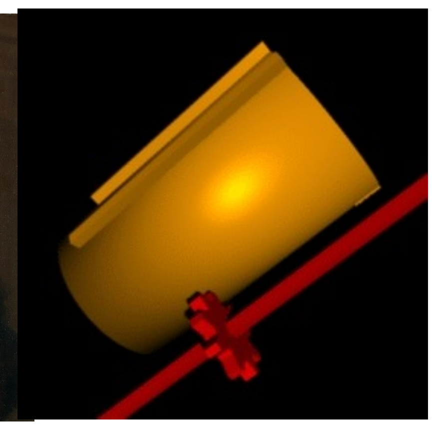
- In 1642, **Blaise Pascal** invented the first practical mechanical calculator, the **Pascaline**, to help his tax-collector father with additions and subtractions. Pascaline used cogs, gear wheels with teeth around their edges.
- In 1671, **Wilhelm Gottfried Leibniz**, built a more advanced machine, the **Stepped Reckoner**, that used a stepped drum, a cylinder with teeth of increasing length around its edge. The Leibniz machine could also multiply, divide, and calculate square roots. It also pioneered the idea of memory storage. **Calculators** used this design for the next three centuries.

Blaise Pascal



Pascaline

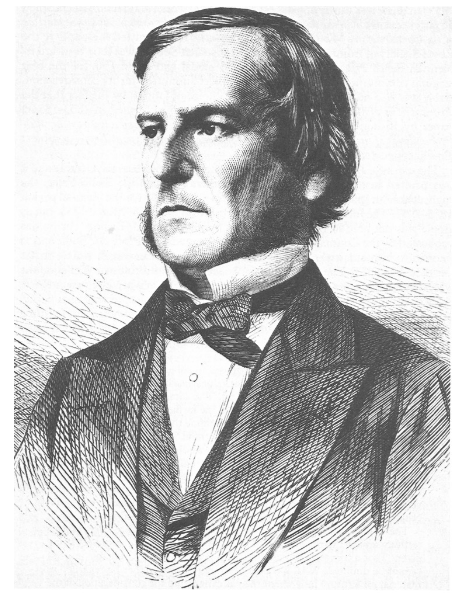
Wilhelm Gottfried Leibniz



stepped drum

Binary arithmetic and Boolean algebra

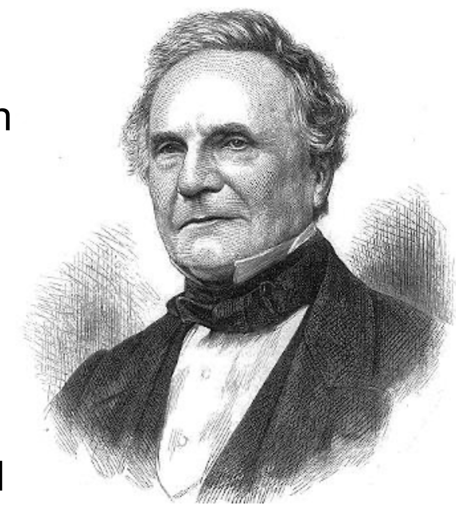
- Leibniz also invented the **binary system** which allows us to represent any decimal number using only the two digits, **zero** and **one**, and described how binary arithmetic, such as addition, subtraction, multiplication, and division can be performed.
- In 1854, **George Boole** used binary numbers to invent a new branch of mathematics called **Boolean algebra** (stay tuned next week!).
- Binary numbers and Boolean algebra are at the heart of modern computers which use them to perform arithmetic and logical operations.



George Boole

The Analytical Engine

- Pascal's and Leibniz's calculators sped up calculations but required a human operator. Instead, computers are machines that operate automatically, without needing human aid, by following instructions written in a program.
- In 1834-36, **Charles Babbage** designed the **Analytical Engine**, the first design for a general computing device that could perform any mathematical calculation.
- It could be used for more than one particular computation, could be given data and run operations in sequence, had memory, branches, looping, and even a primitive printer!
- Babbage's design of the Analytical Engine influenced the first generation of computer scientists. He is considered the "father of the computer."

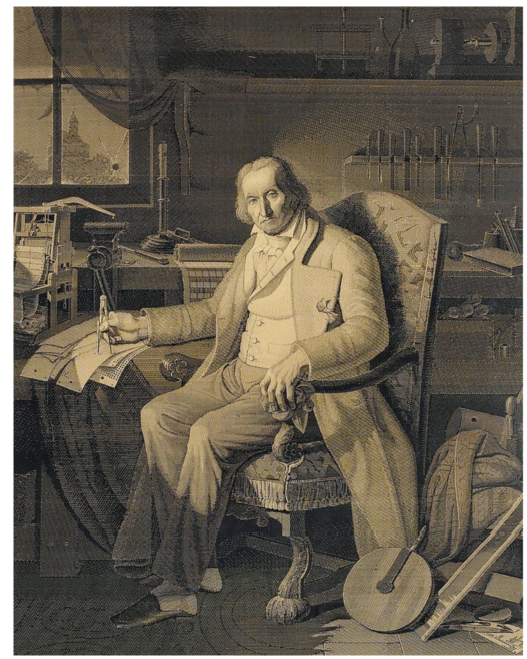


Charles Babbage

Drawing inspiration from Jacquard's looms

Punched cards

- The Analytical Engine could be programmed using **punched cards**, a technology that had been developed to provide instructions for weaving on a mechanical loom in 1805 by **Joseph-Marie Jacquard**.
- These cards could provide input for different weave patterns, to easily produce complex results. Babbage envisioned using them to provide instructions for a program.
- None of Babbage's programmable "engines" were completed during his lifetime due to the scale and cost they required.



Joseph-Marie Jacquard

The first program and programmer

- In 1843, one of Babbage's correspondent's, **Ada Lovelace**, was hired to translate lecture notes on the Analytical Engine from French to English.
- She added extensive notes to this paper with her own thoughts. One of these notes contained an example that showed how the Analytical Engine could be used to calculate Bernoulli numbers.
- This was the first (hypothetical) **program** to be written for a computer, so Lovelace is considered the first **programmer**.

Ada Lovelace

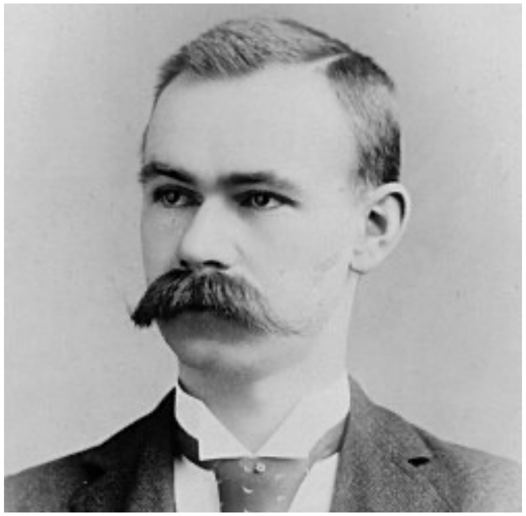
Computational thinking

- Ada Lovelace is also credited as being the first person to realize that **computers could be used for more than just math**. One of her notes read:
- *"[The Analytical Engine] might act upon other things besides number... Supposing, for instance, that the fundamental relations of* ***pitched sounds*** *in the* ***science of harmony and of musical composition*** *were susceptible of such expression and adaptations, the engine might compose* ***elaborate and scientific pieces of music*** *of any degree of complexity or extent."*

Ada Lovelace

Engines of calculation go big

- In the late 1880s, American statistician **Herman Hollerith** built one of the world's first practical calculating machines, which he called a **tabulator**, to help compile census data.
- A U.S. census was taken every 10 years, but the growth of the population meant that it took 7.5 years to tally people by hand!
- Hollerith's tabulating machine tallied the entire census in only six weeks and completed the full analysis in just two and a half years.
- Hollerith set up the Tabulating Machine Company in 1896 which was renamed in 1924 as **IBM** (International Business Machines).



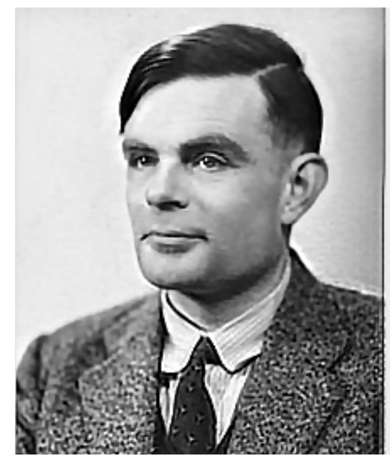
Herman Hollerith

tabulator

A general model of computers

- Actual physical general-purpose computers would not be developed until mid-20th century.
- But just before the first computers were built, in 1936, American mathematician **Alonzo Church** and English mathematician **Alan Turing** developed a general model of what can be computed (by today's computers).
  - This is now referred to as the **Church-Turing Thesis**.
- Turing invented the concept of a 'Turing Machine', a simple abstract machine that manipulates symbols on a tape according to a set of rules.
- It is widely acknowledged today that **all general-purpose computers can be reduced to the idea of a Turing Machine**.
- If a computer is as powerful as a Turing machine, it’s **Turing complete**. Your laptop, phone, microwave, thermostat, are all Turing complete.

Alonzo Church



Alan Turing

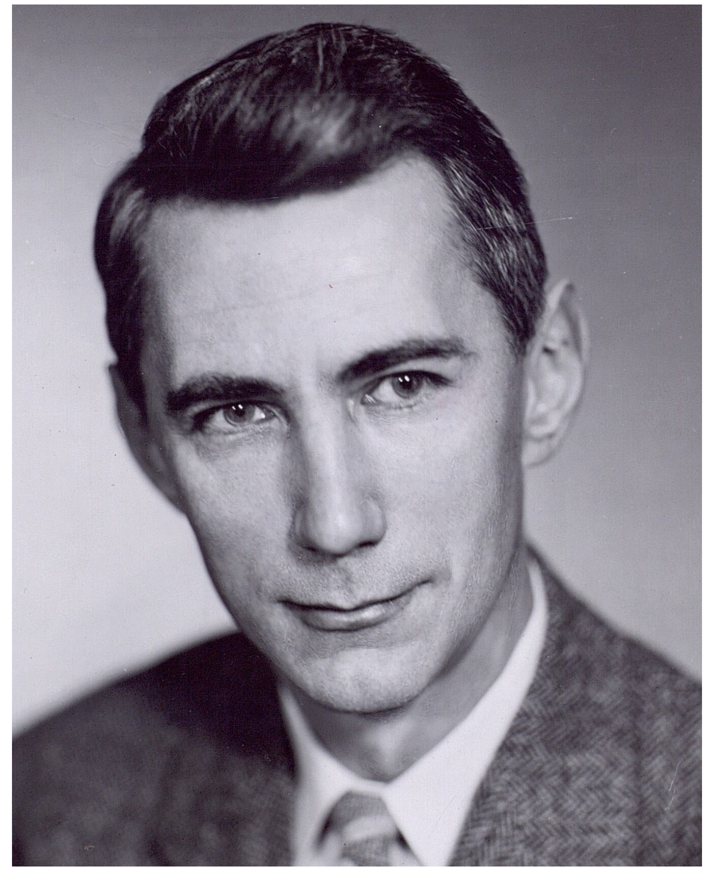
Limits of computation

Kurt Gödel

- Earlier, in 1931, logician **Kurt Gödel** proved the Incompleteness Theorems, which showed that every formal system whose theorems can be listed by an algorithm will have some statements that are unprovable within the system.
- Church used Lambda calculus to prove that Hilbert's Entscheidungs/decision problem is unsolvable.
  - Is there an algorithm that takes as input a statement and provides a yes or no answer so that's always accurate?
- Turing approached the problem differently and demonstrated that not everything is computable by a Turing machine (i.e. a computer) by formulating the **Halting problem**.
  - He used a proof by contradiction to show that it's impossible to write a program that can *always* determine whether another program with a given input will run forever or halt (stop).
  - **This proves that not all problems can be solved by computation!**
- More about theory of computation in CS101.

Circuits

- In 1937, **Claude Shannon** translated Boolean logic into a physical format with electronics.
- This work became the foundation of **circuit design** and made it possible to design the computers we know today.
- Shannon also introduced many of the core ideas of abstraction, encoding, and compression we use today.



Claude Shannon

Computing Devices in World War II

- Shortly after this electronic breakthrough, World War II began. Computing was used to gain advantages in wartime efforts.
- Computing played the most powerful role in code-breaking, as Allied forces attempted to decipher German messages and vice versa.
- The German forces used a device called the **Enigma Machine** to encrypt communications. This encryption used a type of substitution cipher with a shared key. German officers were given key lists ahead of time and would set a new key every day.
- The Allied forces were able to reconstruct the physical device. However, they had to check all possible keys by hand every day, which took too long to be useful. This lasted until someone noticed a pattern in German messages – they always sent a weather report at 6am each day. The common words in this report made it easier to check possible keys computationally.

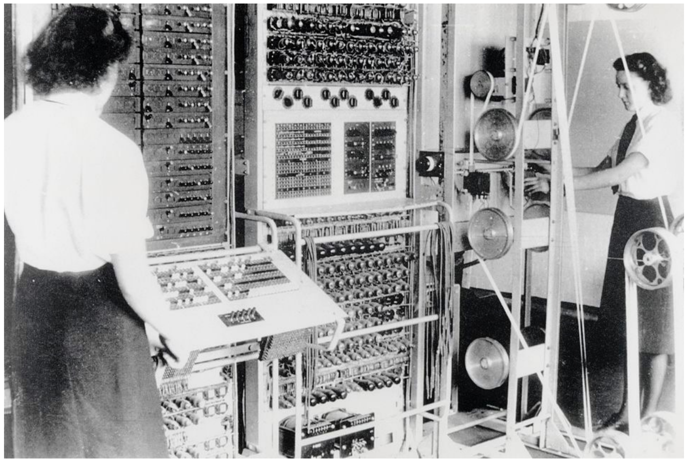
The Bombe and Alan Turing

- The original deciphering machine, the Bomba, was designed by Marian Rejewski in Poland in 1938. Due to improvements in the Enigma and a lack of funds, the idea was passed to Britain.
- In 1939, Alan Turing worked with a team to develop ***the Bombe***, which checked all possible settings to see if they could find one that matched the expected words.
- This process was dramatized in the movie “The Imitation Game.”
- Turing later on was convicted because he was gay and given the choice between imprisonment or probation with hormonal treatment to suppress his sexuality. He chose the latter but took his own life in 1954.
    - The highest distinction bestowed on a computer scientist is the **Turing award**, in honor of Alan Turing.

The Colossus

- Later in the war, German forces started using a new encryption system for high-security messages. The **Lorenz cipher** proved much harder to crack, as the Allied forces had no information about the machine used to produce them.
- From 1943-1945, English engineer **Tommy Flowers** led a team to design the **Colossus**, which was used to break Lorenz ciphers.
- Colossus is widely considered to be the **first electronic programmable computer**. However, it could only be programmed for cipher-breaking, not general tasks.
- As with many war-time inventions, the existence of the Colossus machines was kept secret until the mid-1970s

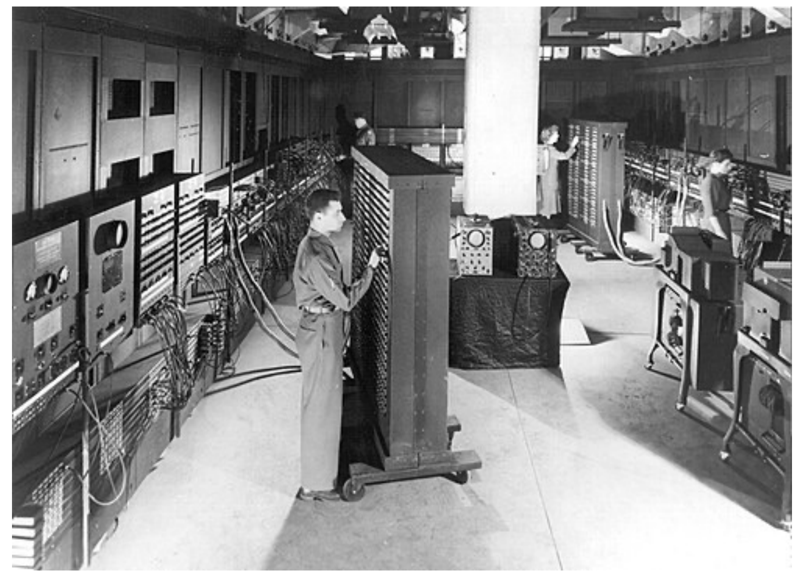
Tommy Flowers



Colossus MK-2

ENIAC - The first modern computer

- In 1945, after the war ended, companies and research groups worked on designing computers for corporate and military use.
- At University of Pennsylvania, **John Mauchly** and **J. Presper Eckert** designed the **ENIAC**, the world's first electronic truly **general-purpose** programmable electronic computer. It influenced many machines that came after it.
- This machine was programmable (by moving wires) and had input and output in the form of punch cards. It could only hold up to 200 decimal digits in memory at first. That's around 80 bytes!
  - ENIAC took 1,800 sq. ft of space and weighed around 30 tons.
- It was operational for half a day at a time due to mechanical failures but led to more computations in 10 years than the entire human race before that!

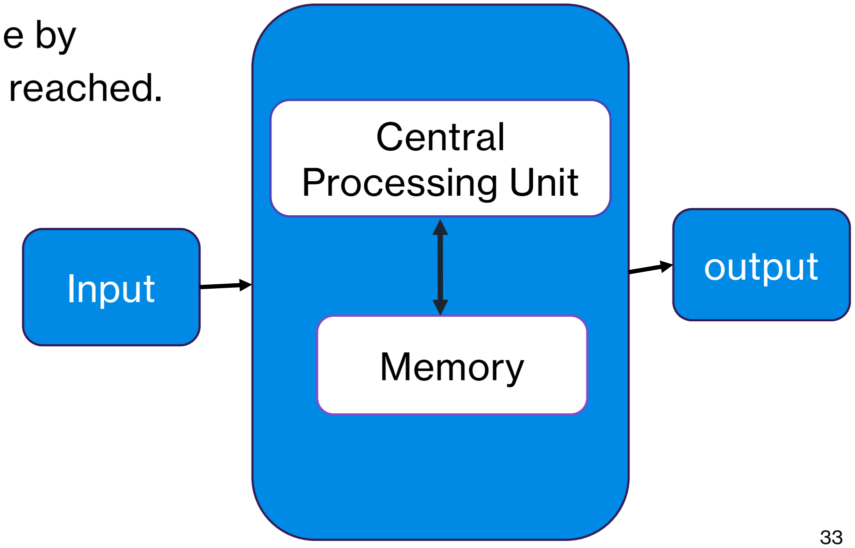


ENIAC

Modern software architecture

- With the introduction of general-purpose computers came the need for software systems to support programming. In 1945, the **software architecture** of computers that we use today was designed.
- **John von Neumann** introduced the **von Neumann architecture**, which organized the CPU, memory, and input/output.
- This also introduced the idea of representing machine code by running instructions sequentially until a conditional jump is reached.

John von Neumann

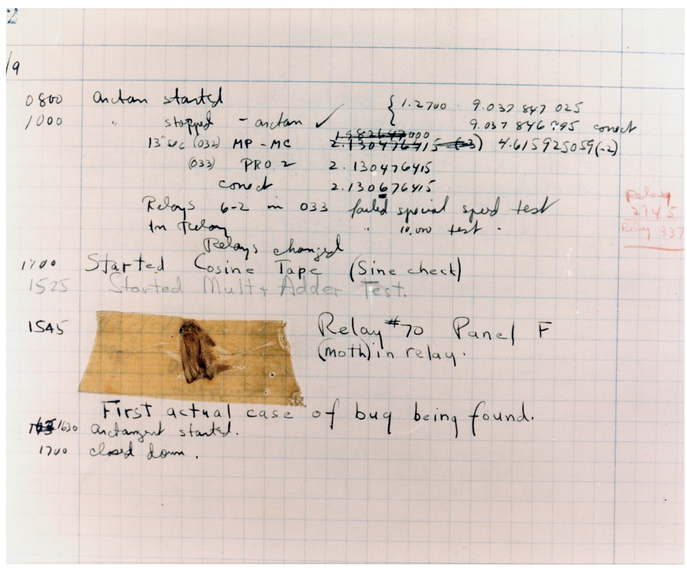


Programming languages

- Computers can only understand **machine language/code** which is sequences of 0s and 1s specific to their machine.
- In early days, translated pseudo-code written in English into machine code.
- By mid-20th century, programmers created reusable helper programs in machine code that read text-based instructions and assemble them into corresponding machine code. These programs are called **assemblers** and they read programs written in an **assembly language** and convert them into native machine code.
- Assembly language was still not versatile enough and higher-level programming languages slowly developed but their instructions still needed to be converted to assembly or machine code.
- Grace Hopper developed the first **compiler**, a program that reads **source code** in a high-level language and translates it into a low-level language like assembly or machine code.
- In 1959, IBM released **FORTRAN** which dominated early programming but could only be compiled on IBM machines.
- In 1959, **Grace Hopper** led the Committee on Data Systems & Languages, that created **COBOL** the first programming language that could be used across different types of machines.
  - This paradigm is known as “Write Once, Run Everywhere”.
- This led to an evolution of higher-level programming languages, e.g., 60s saw ALGOL, LISP, BASIC, 70s had Pascal, C, Smalltalk, 80s brought C++, Objective C, Perl, 90s: Python, Ruby, Java, JavaScript, 2000s, Swift, C#, Go, and so on.

Bugs

- **Grace Hopper** also worked on Harvard Mark II, an early electromechanical computer that used relays. In 1947, operators pulled out a dead moth from a malfunctioning relay. Grace Hopper noted *“from the on, when anything went wrong with a computer, we said it had **bugs** in it."*



The first recorded bug


Grace Hopper

Association of Computing Machinery (ACM)

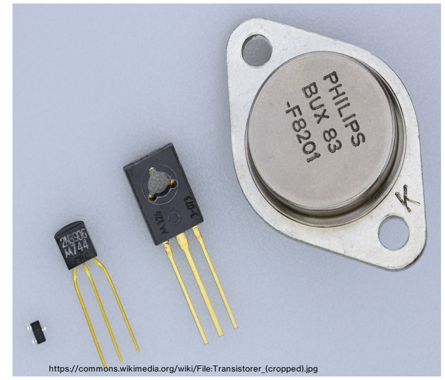
- In 1947, the Association of Computing Machinery (**ACM**) was founded as a U.S.-based international computing society. It has more than 100k members, half outside the U.S.
- It oversees the publishing of journals, sponsoring of conferences, and distribution of awards, like the Turing award.
- As a Pomona student, you have access to the articles published in the ACM Digital Library through the Claremont Colleges Library.
  - If you major or conduct research in computer science, this will be an invaluable tool.
- Other computing societies include the IEEE-CS, AAAI (Association for the Advancement of Artificial Intelligence), International Association for Cryptologic Research (IACR), Association for Computational Linguistics, American Academy of Arts & Sciences (AAA&S), and American Association for the Advancement of Science (AAAS).

Artificial Intelligence (AI)

- Can we create a thinking machine that is intelligent, has consciousness, can learn, has free will and is ethical? Philosophical roots of AI go back centuries.
- Alan Turing in the 1950s devised the “Turing Test” to judge whether a machine was conscious and intelligent raising the possibility of programming a computer to behave intelligently.
- The term AI was coined in 1956 by **John McCarthy** in the Dartmouth Summer Research Project on Artificial Intelligence.
- Artificial Intelligence research has also given birth to related fields like Machine Learning, Natural Language Processing, Computer Vision, Robotics, etc. that tackle different aspects of human intelligence and capabilities.
- **AI winter** cycles are accompanied by new booms. We are currently amidst a boom brought by neural networks and deep learning.

The transistor

- Originally, computers were only used for corporate or government purposes. Individuals did not own computers, because they were far too large and difficult to interact with.
- This changed due to two events: invention of technology that made computers smaller, and invention of interaction modalities that made computers easier to work with.
- In 1947, **John Bardeen**, **William Shockley** and **Walter Brattain** at AT&T Bell Labs designed the **transistor**. This device could be used to switch electric signals.
- Previously, computers had to use vacuum tubes, which were very large. The invention of the transistor made it possible to make computers smaller.

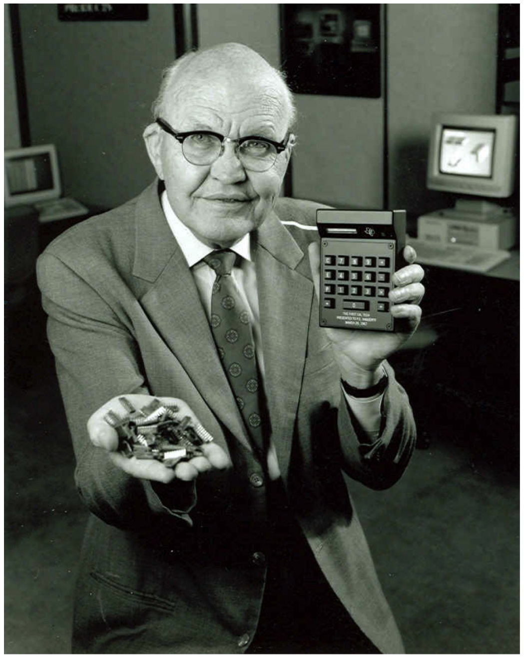


transistors

John Bardeen
William Shockley
Walter Brattain

The integrated circuit

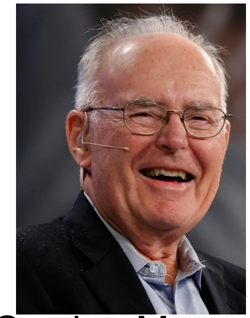
- In 1958, **Jack Kilby** invented the **Integrated Circuit** (IC). This is a small electronic device (or 'chip') that can contain many circuits and is easy to produce. It was possible to make ICs because of the invention of the transistor.
  - Kilby also co-invented the handheld calculator.
- A few months later, **Robert Noyce** made ICs practical by building them out of the abundant and stable **silicon** instead of the rare and unstable germanium Kilby used.
  - Noyce was nicknamed “the Mayor of Silicon Valley”
- The IC again made it possible to make computers much smaller, as more electronics could be fit onto a smaller surface.



Jack Kilby holding ICs

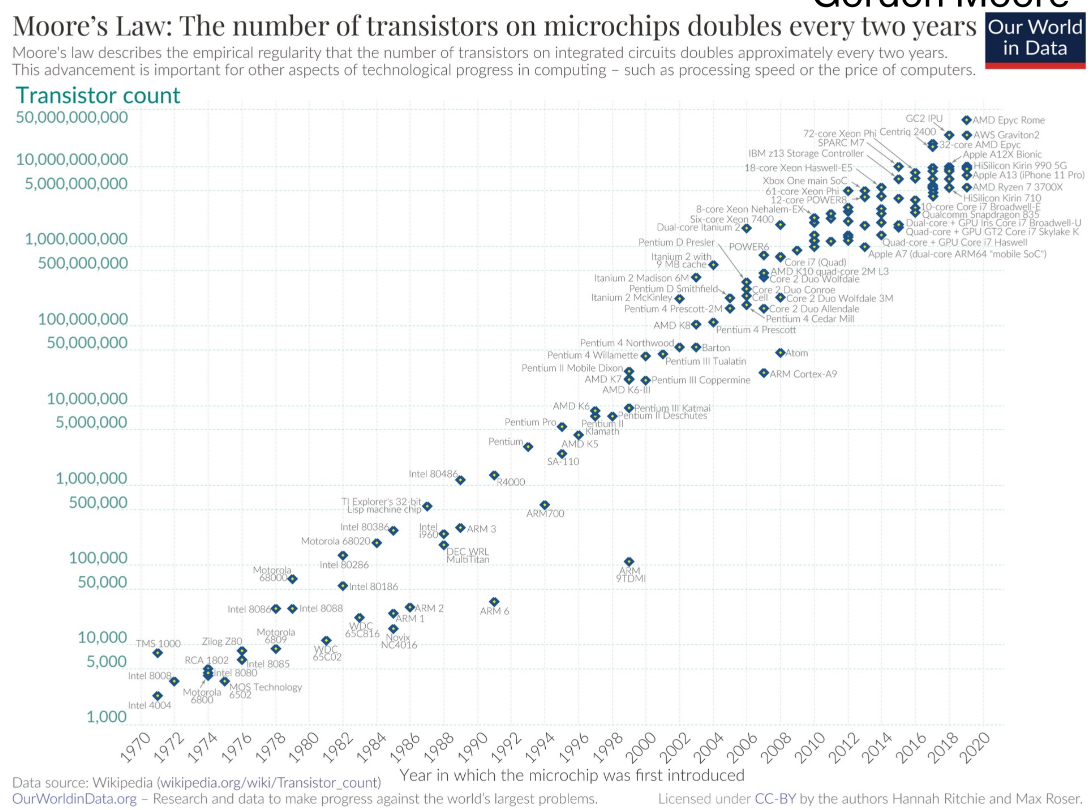
Robert Noyce

Moore's Law and The Microprocessor



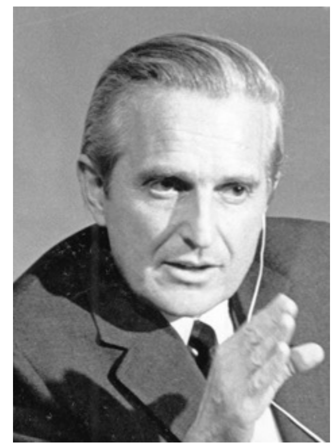
Gordon Moore

- In 1965, **Gordon Moore** introduced the **Moore's Law**, a business model which states that “*the number of transistors on an IC doubles every two years*.”
- By 1971, this led to the invention of the **microprocessor** at **Intel** (which was co-founded by Moore and Noyce). A microprocessor is a whole processor that can fit onto a single chip.
- This breakthrough made it possible to put chips in many new devices, like calculators and clocks.



The mother of all demos

- In 1968, **Douglas Engelbart** presented work he had done at Stanford to a group of engineers at a computer conference. This presentation later became known as the **Mother of All Demos** because it introduced an astounding number of technologies that we use to this day.:
  - The computer mouse
  - The GUI (Graphical User Interface)
  - The WYSIWIG (What You See Is What You Get) text editor
  - The concept of multiple windows
  - Revision control
  - Video conferencing
  - Real-time collaborative editing
- You can watch the demo for yourself online: https://www.youtube.com/watch?v=yJDv-zdhzMY



Douglas Engelbart

Commercially successful personal computer

- It is hard to pinpoint the exact time that personal computers made an appearance since there are multiple contenders.
- But in 1975, MITS released **Altair 8800**, the first commercially successful personal computer. Priced about $2000 today's dollars, it came as a built-it-yourself kit.
- Tens of thousands of kits were sold to computer hobbyists and soon accessories were sold, leading to a rise of a movement of computer enthusiasts.
- The most famous one is the **Homebrew Computer Club** which first met in 1975.

Altair 8800

Computing companies

- In 1975, **Bill Gates** and **Paul Allen** founded **Microsoft**.
- They convinced MITS if Altair 8800 programs were written in BASIC. To do so, they created an **interpreter**, a program that translated code written in BASIC to low-level machine code.
  - Interpreters translate as the program runs instead of beforehand, as compilers do.
- At the first meeting of the Homebrew Computer Club, **Steve Wozniak** was so inspired by Altair 8800 that he set to create his own personal computer. In 1976, he demonstrated his prototype to the Club. It could connect with a TV and included a text interface.
- Interest was high. Fellow Club member **Steve Jobs** convinced Wozniak to sell an assembled motherboard (you still needed to add a keyboard and monitor) instead of sharing the designs for free. It was sold as **Apple I** and Steve Jobs and Steve Wozniak founded **Apple**.

Bill Gates and Paul Allen

Steve Wozniak and Steve Jobs with Apple I

The 1977 trinity and open/closed architecture

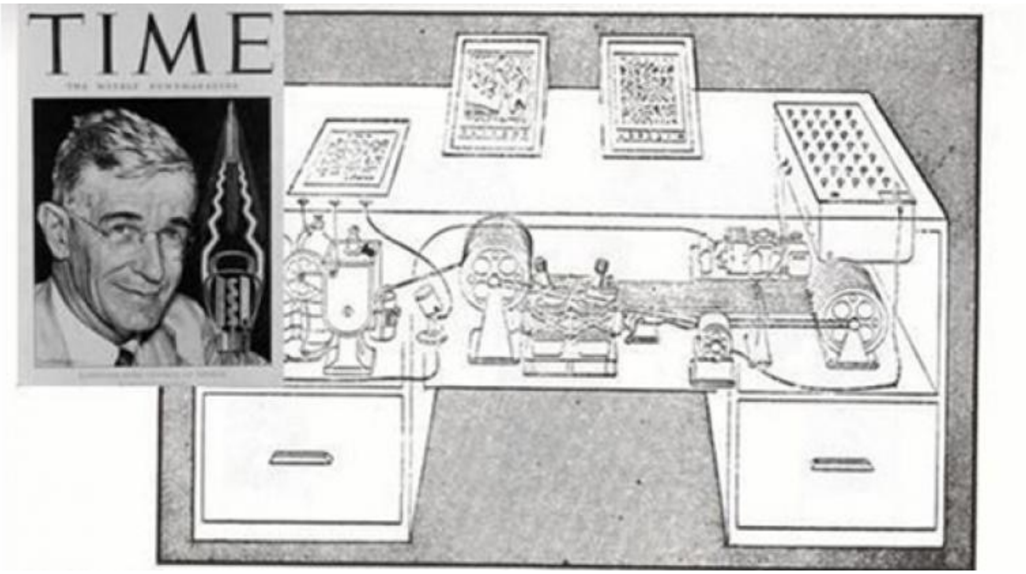
- Apple I was also sold like a kit which appealed to hobbyists and tinkerers but not to the masses. This changed with three computers known as the **1977 trinity**.
  - The first, **Apple II**, was professionally designed and manufactured, offering color graphics and sound. Millions of computers were sold, propelling Apple at the forefront of the personal computer industry.
  - The second, **TRS-80 model I**, was less sophisticated but sold at half-price of Apple II.
  - The third, **Commodore PET 2001**, combined computer, monitor, keyboard, and tape drive into one device.
  - All three came with BASIC interpreters which allowed less technical audience to create programs and targeted households, small businesses, and schools.
- IBM took notice and designed the IBM PC which used Microsoft's operating system **MS-DOS** and offered an **open architecture** with expansion slots, allowing third parties to create hardware and peripherals like graphics and sound cards, joysticks, external hard drives, etc.
  - **IBM compatible** computers took over most of the market. IBM's approach was in contrast with Apple's **closed architecture**. This led to the rise of the "**MAC vs PC**" debate.

Graphical user interface (GUI)

- After the Mother of All Demos, several people on Engelbart's team went to work at **Xerox PARC**, to further develop the concepts into the first GUI computer, the **Xerox Alto**.
- Xerox established the **desktop metaphor** that emulated one's desk on a 2D screen and the **WIMP interface** (windows, icons, menus, pointer) along with **buttons**.
- In 1979, Apple employees were invited to PARC and were shown their GUI implementation. Steve Jobs said:
  - *"It was like a veil being lifted from my eyes. I could see the future of what computing was destined to be."*
- They implemented similar ideas into the Apple Lisa (1983, commercial flop) and ultimately into the **Apple Macintosh** (released in 1984) to great acclaim.
- In 1981, Microsoft visited Apple and helped them develop some apps. They took the GUI idea from Apple and used it in their first operating system, MS-DOS, released in 1985.
  - A rivalry between Microsoft and Apple would start then, with Microsoft dominating 95% of personal computers.

Ideating the Internet

- Some of the core concepts of how the Internet would work were introduced well before it was implemented.
- In 1945, **Vannevar Bush** published As We May Think on the Atlantic, which envisioned a system (**Memex**) to aid in research work. Bush invented the concept of **hypertext** (or link)!
    - *"Consider a future device... in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to his memory."*



Vannevar Bush and Memex

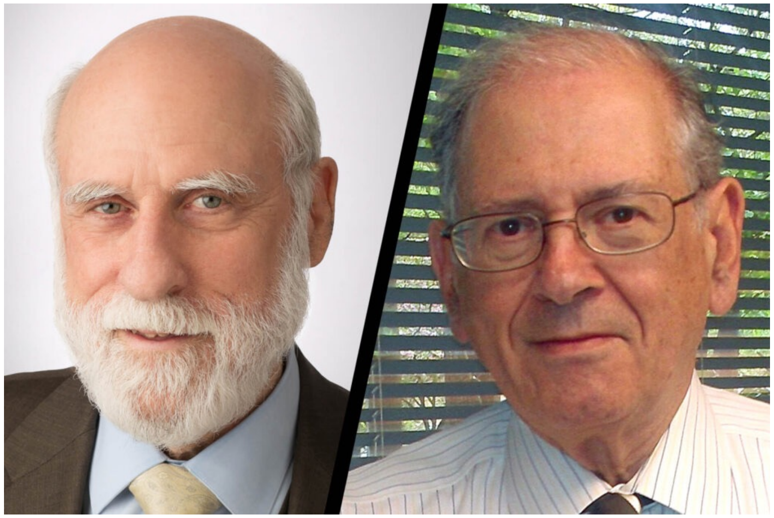
ARPANET

- In 1969, the US military wanted to create a decentralized communication system so that communications could not be knocked out entirely by a nuclear attack.
- DARPA (Defense Advanced Research Projects Agency) collaborated with several universities to build the **ARPANET**, the Advanced Research Projects Agency Network.
- First connection between UCLA and Stanford, but it grew quickly.
- The first email was sent over ARPANET in 1971!

ARPANET

Communication protocols

- In 1982, **Vinton Cerf** and **Robert Kahn** designed and advocated for the **TCP/IP** protocol.
- TCP organizes data that is being sent between computers; IP delivers that data to the correct destination (based on IP addresses).
- The invention of TCP/IP made it much easier to connect computers together, which helped ARPANET expand its reach.
- Because of this, Vint Cerf and Bob Kahn are known as the “fathers of the Internet.”
- By 1984, the US military broke off from ARPANET to form their own private network, MILNET.
- More organizations and companies started to join the public network, forming the **Internet** as we know it.



Vint Cerf and Robert Kahn

1990s: the World-Wide Web

- In 1989, **Tim Berners-Lee** invented a new language, **HTML** (HyperText Markup Language), and a new notation, **URL**, that would revolutionize how people communicated over the Internet.
- Berners-Lee also created the first **web browser** and **web server**. This led to the beginning of **websites** as we know them and the creation of the **World-Wide Web**, an information system that enables content sharing over the Internet.
- For this contribution, he is known as “the father of the World-Wide Web.”
- The first webbrowser that allowed graphics to be embedded along text was **Mosaic**. Others, like Netscape Navigator, Internet Explorer, Opera, Mozilla, followed.
- New websites popped up continuously with the most famous one being **Yahoo!**.
- Search engines also started popping up in the 1990s. **Google** wasn't founded until 1998, and **Wikipedia** wasn't created until 2001!

2000s: social media and cloud computing

- As more people got on the Internet, **social media networks** started to pop up.
- Some started in the late 90s with MySpace being the first truly global one.
- Of the current big networks, LinkedIn started in 2003, Facebook in 2004, and Twitter/X in 2006.
- **Cloud Computing** also started in the 2000s. Amazon's Elastic Compute Cloud started in 2006; Microsoft Azure started in 2008.

2010s: smartphones, tablets, and autonomous vehicles

- The growth of the Internet and the desire to remain connected led to portable computing devices.
- Smartphones first appeared in 2007 with iPhone and gained widespread popularity in the 2010s.
- Tablets also became popular in this timeframe.
- Autonomous robots and self-driving vehicles with intelligent sensors became more mainstream.

Steve Jobs holding an iPhone

2020s: XR, artificial intelligence, and ?

- Although, still mid-way, 2020s already have left their mark on the history of computing.
- Big companies are pushing for **XR** which encompasses, **augmented reality** (e.g., Pokemon Go) and **virtual reality** (e.g., Meta Quest and Apple Vision Pro) to become the new computing paradigm.
- Companies like Google, Open AI, Anthropic are advancing and commercializing **artificial intelligence** at a frantic pace (e.g., ChatGPT, DALL-E, Midjourney).
  - We will learn more about AI at the end of the course.
- What will the next five years bring?

Ethics of CS

Ethics

- Ethics can be organized in three levels:
  - **Personal ethics:** an individual's moral principles and values guiding their behavior. It's shaped by family, culture, and personal experiences.
  - **Professional ethics:** the ethical standards and principles specific to a particular profession or field. It often includes codes of conduct and guidelines designed to ensure responsible and ethical practice within that profession.
  - **Societal ethics:** the ethical principles that govern how a society functions, including laws, customs, and social norms. It reflects the collective values and expectations of a community or culture.
- When we move from the theoretical concepts of computer science to applying those theories in real life, the decisions we make have consequences.

Activity: ethics for computing professionals

- What ethics should govern the computing profession?

ACM Code of Ethics

- ACM has established a Code of Ethics and Professional Conduct (known as "the Code") that expresses the conscience of the computing profession.
- The latest Code was adopted in 2018 and is split into four sections.
  - Section 1: fundamental ethical principles that form the basis for the remainder of the Code.
  - Section 2: additional, more specific considerations of professional responsibility.
  - Section 3: guides individuals who have a leadership role, whether in the workplace or in a volunteer professional capacity.
  - Section 4: principles involving compliance with the Code. Commitment to ethical conduct is required of every ACM member and award recipient.

1. General Ethical Principles

A computing professional should...

1. Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing.
2. Avoid harm.
3. Be honest and trustworthy.
4. Be fair and take action not to discriminate.
5. Respect the work required to produce new ideas, inventions, creative works, and computing artifacts.
6. Respect privacy.
7. Honor confidentiality.

2. Professional Responsibilities

A computing professional should...

1. Strive to achieve high quality in both the processes and products of professional work.
2. Maintain high standards of professional competence, conduct, and ethical practice.
3. Know and respect existing rules pertaining to professional work.
4. Accept and provide appropriate professional review.
5. Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.
6. Perform work only in areas of competence.
7. Foster public awareness and understanding of computing, related technologies, and their consequences.
8. Access computing and communication resources only when authorized or when compelled by the public good.
9. Design and implement systems that are robustly and usably secure.

3. Professional Leadership Principles

A computing professional, especially one acting as a leader, should...

1. Ensure that the public good is the central concern during all professional computing work.
2. Articulate, encourage acceptance of, and evaluate fulfillment of social responsibilities by members of the organization or group.
3. Manage personnel and resources to enhance the quality of working life.
4. Articulate, apply, and support policies and processes that reflect the principles of the Code.
5. Create opportunities for members of the organization or group to grow as professionals.
6. Use care when modifying or retiring systems.
7. Recognize and take special care of systems that become integrated into the infrastructure of society.

4. Compliance with the Code

A computing professional should...

1. Uphold, promote, and respect the principles of the Code.
2. Treat violations of the Code as inconsistent with membership in the ACM.

Case study: malware disruption

Before we proceed, some vocabulary:

- **Malware**: software that is intentionally designed to cause disruption to computers, leak private information, gain unauthorized access to computer systems, or deprive access to information.
  - Common subtypes: computer viruses, botnets, **worms**, Trojan horses, ransomware, spyware, etc.
- **Spam**: unsolicited messages sent to a large numbers of recipients for the purpose of commercial advertising, non-commercial proselytizing, or any illegal purpose.
- **ISPs** (Internet Service Providers): organizations that provide services related to accessing the Internet. You might know some: Verizon, Frontier, Spectrum, Cox, AT&T, etc.

Activity: malware disruption

- *Rogue Services* advertised its web hosting services as *"cheap, guaranteed uptime, no matter what."* While some of Rogue's clients were legitimate web-based retailers, the majority were focused on **malware** and **spam** and used Rogue Services' reliability guarantees to protect their illegal operations.
- Despite repeated requests from major ISPs and international organizations, Rogue Services refused to intervene with these services, citing their "no matter what" pledge to their customers. Furthermore, international pressure from other governments failed to induce national-level intervention, as Rogue Services was based in a country whose laws did not adequately proscribe such hosting activities.
- Ultimately, Rogue Services was forcibly taken offline through a coordinated effort from multiple security vendors working with several government organizations. This effort consisted of a targeted worm that spread through Rogue Services' network. All of Rogue Services' clients were affected and much of the data stored with the ISP in the process. No other ISPs reported any impact as it was designed to not spread further. As a result of this action, malware circulation decreased.

Analysis: malware disruption – for Rogue Services

- Rogue Services' actions include violations of several principles of the ACM Code of Ethics.
- By allowing for the hosting of malware, they facilitated the harm caused by their clients, violating both Principles 1.1 (Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing) and 1.2 (Avoid Harm).
- Additionally, they were complicit in violating Principle 2.8 (Access computing and communication resources only when authorized or when compelled by the public good), as the ISP was aware that their machines were hosting code that caused infections that were clearly not authorized.
- Finally, Rogue failed to consider the public good, violating Principle 3.1 (Ensure that the public good is the central concern during all professional computing work).

Analysis: malware disruption – for worm authors

- Key nuance of Principle 1.2 (Avoid Harm). Given that the worm was designed to cause harm to Rogue Services' systems, the authors were obligated to ensure the harm was ethically justified. The worm aimed to shut down services that were harmful and malicious, an intent consistent with the moral obligations identified in Principle 1.1 (Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing).
- Additionally, the worm included mechanisms to limit itself solely to Rogue Services' systems, thus demonstrating an attempt to minimize unintended harm. Rogue's retailer clients could rightfully object to the deletion of their data, so a better solution would have included additional precautions to avoid this unintentional harm.
- The worm also highlights the guidance in Principle 2.8 (Access computing and communication resources only when authorized or when compelled by the public good.) The worm accessed Rogue Services' systems without authorization, destroying data in the process. However, the goal of targeting malware demonstrates the service disruption was consistent with the public good.

Professional ethics in our field

- Software engineers don't have general licensing requirements in contrast to professions like lawyers, doctors, and engineers.
- Being a computer science researcher and analyzing data generally does not require approval by a review board (IRB), unless it includes humans.
- There is no single regulatory body and consequences are limited.
- We will next see examples of Ethics in Computer Science split across four categories: we will mostly focus on algorithmic bias and data collection and privacy, and touch upon AI and autonomous systems, and impact on environment.

Algorithmic Bias

Algorithmic decision-making and bias

- Algorithmic bias describes systematic and repeatable harmful tendency to create unfair outcomes, such as "privileging" one category over another in ways different from the intended function of the algorithm.
- Sometimes, bias is due to the algorithm design, the unintended or unanticipated ways the data were collected and used to train the algorithm.
- Algorithmic bias can reinforce social biases of race, gender, sexuality, and ethnicity.
- It can also result to privacy violations.
- Algorithms are often seen as neutral and unbiased which can make them falsely appear as more unbiased than humans and more authoritative.
- We will see a number of examples of bias in public and private settings.

Wrongfully accused by an algorithm

- In January 2020, Robert Julian-Borchak Williams was in his office when he got a call from the Detroit Police Department telling him to come to the station to be arrested. He thought it is a prank.
- He was later arrested at home and drove to a detention center where he had his mug shot, fingerprints and DNA taken, and was held overnight.
- He was accused of shop-lifting five watches from an upscale boutique, being shown a surveillance video as “proof” that it was him who committed the crime.
- Mr. Williams was wrongfully arrested based on a flawed match from a facial recognition algorithm.
- Studies have shown that while the technology works relatively well on white men, the results are less accurate for other demographics, partly because of a lack of diversity in the images in the databases.
- Mr. Williams, who is Black, was held in custody for hours although the match with the suspect was obviously wrong. When the case was called, the prosecutor moved to dismiss, but “without prejudice,” meaning Mr. Williams could later be charged again.
- NY Times ran a popular story which resulted in the case and fingerprint data expunged.

Machine bias in risk assessments in criminal sentencing

- In 2016, ProPublica released an analysis on how the software COMPAS, Correctional Offender Management Profiling for Alternative Sanctions, assigned recidivism scores.
- Scores like this-known as risk assessments-are increasingly common in U.S. courtrooms. They are used to inform guiding judges during criminal sentencing.
- The journalists obtained the COMPAS risk scores assigned to more than 7,000 people in Florida and checked how many were charged with new crimes over the next two years.
- The scores proved remarkably unreliable in forecasting violent crime: Only 20% of the people predicted to commit violent crimes actually went on to do so.
- The formula was particularly likely to falsely flag Black defendants as future criminals, wrongly labeling them this way at almost twice the rate as white defendants.
- White defendants were mislabeled as low risk more often than Black defendants.

https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Search engines

- Social scientist Dr. Safiya Noble showed in her 2018 book, Algorithms of Oppression: How Search Engines Reinforce Racism, examples of algorithmic bias when she used search engines like Google.
  - Terms like "black girls" returned pornography and "Jew" returned anti-Semitic pages.
  - Google claimed it was unable to erase those pages unless they were considered unlawful.
- Google similarly received heat when queries related to occupations were found to propagate sexist stereotypes, e.g., the term CEO or doctor would return pictures with men, while the term nurse would return women.
- Although Google claimed that they fixed the problem, researchers from UW showed that simple tweaks, like quering “CEO + United States” returned fewer photos of cis-female presenting people. Taking a ‘whack-a-mole’ approach doesn’t seem to be fixing the problem.

AI and hiring bias

- Similar biases of racism, sexism, and ageism have been identified in how automatic job application screening tools would rank job applicants' names according to perceived race and gender.
- The boom of large-language models like ChatGPT has led to a "tsunami" of AI-generated resumes that inundate employers.
- Companies have responded with more automation, using AI-ran chats or video interviews.
- But candidates can also use AI to cheat in these interviews.
- Do we live in the AI vs AI era?
- https://www.nytimes.com/2025/06/21/business/dealbook/ai-job-applications.html

AI hallucinations and human hallucinations

- The arrival of ChatGPT and similar tools has transformed not just the hiring process.
- But reasoning systems from OpenAI, Google, and DeepSeek are generating more errors, not fewer.
- Since these AI tools learn from using complex mathematical systems to analyze enormous amounts of digital data and they cannot distinguish what is true or not, they often make things up, a phenomenon known as **AI hallucinations**.
- Beyond the dangers of inventing false information, they also spread conspiracies which can harp on people's mental health when blindly trusting information from seemingly authoritative systems. This is known as **algorithmic appreciation.**

https://www.nytimes.com/2025/08/26/technology/chatgpt-openai-suicide.html

https://www.nytimes.com/2025/05/05/technology/ai-hallucinations-chatgpt-google.html

https://www.nytimes.com/2025/06/13/technology/chatgpt-ai-chatbots-conspiracies.html

Data Collection and Privacy

Data collection

- Many websites are funded by advertising. They provide content or functionality to consumers for free, and the cost of creating and maintaining the website is covered by advertisers, who pay to have the website show ads to the consumers.
- Advertisers want to show ads only to users who are most likely to buy the product or service.
- Therefore, data about consumers are valuable, because they let advertisers direct ads most effectively.

Profiles

- Advertisers want profiles of consumers. A profile is a collection of facts, including:
  - Demographics: age, gender, race, etc.
  - Market participation: income level, location
  - Preferences: family information, taste in books and movies, favorite restaurants or travel destinations, etc.

Data Economy – Data Collection

- Websites have a strong incentive to get the best data possible on their users, so that they get paid more for advertisements. This has led to **hyper-targeting** in ads, with ads attempting to reach more niche populations.
- Check out the different categories popular websites have identified as relevant for you:
  - Google: https://myadcenter.google.com/home
  - Instagram: https://accountscenter.instagram.com/ad_preferences/ad_topics
  - TikTok: Settings and Privacy > Ads > How your ads are personalized

- *What kinds of data are you comfortable having collected by companies?*
- *Where might you draw a line?*

Third parties

- When we talk about data collection and privacy, we're usually really talking about **data sharing**.
- When a consumer interacts with a company, and a different company learns something about the interaction, we say that the data has been shared with a **third party**.
- This kind of data sharing is easy if both companies know a unique piece of information about the user, like the email address they used when creating an account.
- But people don't always make accounts, and sometimes they use different email addresses.
- This means that the core of data sharing is **tracking**: finding a way to link together all the actions that a person takes on different websites and at different times.

Cookies

- Tracking often relies on **cookies** to identify the people visiting a website. A cookie is a piece of data that a website asks a browser to remember. For example:

1. You visit a website and select the “dark mode” view. Next time you visit it, your browser sends the cookie to the website and it shows you the dark mode view automatically.
2. You visit a shopping website, and, behind the scenes, it assigns you an ID number and tracks which products you looked at. It doesn’t know your name or email address, so it uses the ID in place. Weeks later, you visit the site again and enter your email address to place an order. Your browser sends the cookie, and the company links your email address to the products you looked at the first time you were on the site.
3. **Third-party cookie**: You visit a website that displays an ad. When you visit a different website that displays ads from the same company, your browser sends the cookie, and the ad company updates their records with the new information. If lots of websites show ads from the same ad company, they can see a lot of what you do online.

Fingerprinting

- First-party cookies are usually used for preferences, remembering that a user is logged in, and so on. Third-party cookies are mostly used for tracking, so some browsers allow users to block third-party cookies.
- In response, data aggregators began using **fingerprinting** to track users across websites.
- Fingerprinting is a technique that gathers lots of little pieces of information about the computer visiting a website. With enough little pieces, the website can uniquely identify the computer next time it visits.
- This is possible because your browser makes lots of information about your computer (or phone) visible to the websites you visit.
- This is not done maliciously – services can use this information for good, e.g., knowing the size of your screen lets a website show you the mobile or desktop version of the site.
- Check out the data your browser shares here: https://webkay.robinlinus.com/

Trackers

- **Trackers** are pieces of code created and published by data aggregators: companies that want to collect lots of data on lots of people.
- Other companies use the tracker’s code while building their website. The tracker adds some functionality to the website, like letting the company see which pages of their website get the most traffic.
- At the same time, the tracker reports back to the data aggregator that made it with information about the people visiting the website. Trackers can also be included in emails.
- Trackers use a combination of cookies and fingerprinting to identify the website’s visitors.

Consent

- Regulators around the world have tried a variety of methods to balance the desire of some consumers for privacy with the desires of companies.
- In the US, a model called **notice and choice** has historically been dominant.
- Under the notice and choice model, a company can legally use and share data about a consumer if:
- They **notify** the consumer, usually in a privacy policy or terms of service document, and
- The consumer **chooses** to agree, either by checking an “I agree” box or by continuing to use the website.

- *When a website shows you a privacy policy or terms of service document, what do you do?*

Data privacy regulation

- In the European Union, the General Data Protection Regulation (GDPR) similarly restricts some data sharing and gives consumers additional rights.
- The US has no nationwide data privacy law, but legislators, regulators, and civil society groups have all shown interest in possible future legislation.
- Instead, individual states, about 20 including California which has the California Consumer Privacy Act (CCPA), give consumers rights to know about data collection and reject some collection.

Facebook–Cambridge Analytica data scandal

- In the 2010s, personal data belonging to millions of Facebook users was collected by British consulting firm Cambridge Analytica for political advertising without informed consent.
- The data was collected through an app called "This Is Your Digital Life", developed by data scientist Aleksandr Kogan, a data scientist at the University of Cambridge in 2013.
- The app consisted of a series of questions to build psychological profiles on users, and collected the personal data of the users' Facebook friends via Facebook's Open Graph platform.
- The app harvested the data of up to 87 million Facebook profiles.
- Cambridge Analytica used the data to analytically assist the 2016 presidential campaigns of Ted Cruz and Donald Trump.

https://en.wikipedia.org/wiki/Facebook-Cambridge_Analytica_data_scandal

Protecting your data

- If you want to protect your data online, you have options! Most browsers let you block cookies and can request that websites do not track you. You can also restrict permissions given to websites and applications on your devices.
- You can check what kinds of trackers your browser stops and what your fingerprint looks like here: https://coveryourtracks.eff.org/
- You can see what trackers and fingerprinting techniques a website is using by entering it here: https://themarkup.org/blacklight
- One factor that fingerprinting uses is your IP address. You can hide your IP address from the websites you visit using a VPN. Pomona has a VPN (though then Pomona will know which websites you're accessing): https://www.pomona.edu/administration/its/services

AI and autonomous systems

Deepfakes

- **Deepfakes** are media (images, videos, audio) that have been altered or fabricated using AI to appear as they are real.
- They can have some benign applications in entertainment. BUT!
- They have been used to spread disinformation, incite political divisions, in child sexual abuse material. revenge porn, bullying, and financial fraud.
- Some states are moving to ban their malicious use. E.g., https://www.nytimes.com/2025/05/22/business/media/deepfakes-laws-free-speech.html

Advanced Driver Assistance Systems (ADAS)

- **Level 0 - No automation**. Driver has 100% control
- **Level 1 – Driven assistance**. At least one driver support system that provides assistance, for example, adaptive cruise control.
- **Level 2 – Partial automation**. Can take control over steering, acceleration, breaking but driver should remain active and supervise.
- **Level 3 – Conditional automation**. Uses various driver assistance functions and AI to make decisions but driver must be present, alert, and should be able to take over.
- **Level 4 – High automation**. No human supervision is needed. Does not require a steering wheel or pedals and is designed to stop at system failure. Companies like Waymo or Cruise.
- **Level 5 – Full automation.** Vehicle can drive independently without any restriction. Not available yet.
- Level 3-5 are known also as Automated Driving Systems (ADS) and distinguished by 1-2 (ADAS).

Accidents and fatalities (2019-2024)

- Tesla (ADAS) has had the highest number of incidents (2146). Waymo (ADS) follows next with 415. California is the state with the most self-driving incidents, followed by Texas and Arizona, respectively.
- 10% of autonomous vehicle accidents have resulted in injury, and 2% have resulted in a fatality.
- 83 fatalities related to autonomous vehicle accidents as of June 17, 2024. Below are a few select incidents. For example,
    - In 2019, Walter Huang dropped his child off at school and then engaged the autopilot feature of his Tesla Model X. The car veered out of the lane and began to accelerate, crashing into a barrier at 70mph and killing the driver.
    - In March 2018, an Uber self-driving test vehicle struck and killed Elaine Herzberg, a pedestrian, in Tempe, Arizona. This was the first recorded case of a pedestrian fatality involving a fully autonomous vehicle. The backup driver of the vehicle was later charged with negligent homicide.

https://www.craftlawfirm.com/autonomous-vehicle-accidents-2019-2024-crash-data/

Impact on Environment

Mining for technology

- The Interior's U.S. Geological Survey published a list of 35 mineral commodities considered critical to the economic and national security of the United States.
- These minerals are used virtually in every sector of the economy and type of technology.
- They require mining the Earth, often leading to both environmental and geopolitical problems for the neighboring communities.
- **Conflict minerals**, that is natural resources extracted in conflict zones and then sold to fund the conflict are inextricably linked with the production of computers.
- But even locally, mining can be controversial. For example, lithium mines exist in Nevada and tension exists between environmentalists and green-energy advocates.
- The average smartphone life span is less than 5 years!

https://www.usgs.gov/news/national-news-release/interior-releases-2018s-final-list-35-minerals-deemed-critical-us
https://www.nytimes.com/2025/01/24/magazine/nevada-lithium-mines.html

Powering technology

- A **data center** is a temperature-controlled building that houses computing infrastructure, such as servers, data storage drives, and network equipment.
- Scientists have estimated data centers consumption is at the same level with whole countries, being in the top 10 consumers!
- Data centers are used to train and run the deep learning models behind popular tools like ChatGPT and DALL-E. While not all data center computation involves generative AI, the technology has been a major driver of increasing energy demands.
- In 2021, it was estimated that to train GPT-3, the power used was equivalent to 120 average U.S. homes for a year and that it generated about 552 tons of carbon dioxide.
- The strain on water access to local communities near data centers is real.
- The cost of running an AI vs traditional query can be 10-50x higher!

https://news.mit.edu/2025/explained-generative-ai-environmental-impact-0117

https://www.nytimes.com/2025/07/14/technology/meta-data-center-water.html

Further reading

- Online
  - https://criticallyconsciouscomputing.org/history
  - https://www.explainthatstuff.com/historyofcomputers.html
  - https://www.computerhistory.org/timeline/

- Books
  - Introduction to the History of Computing by Gerard O' Regan
  - Atlas of AI : power, politics, and the planetary costs of artificial intelligence by Kate Crawford

- Acknowledgments
  - Some slides have been adapted from material from CMU CS110 – Principles of Computing. Shared by permission by instructor Dr. Kelly Rivers.