

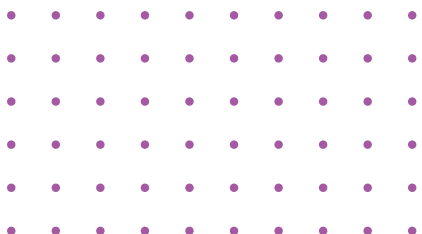
# WEEK 1 - PART 1

## INTRO TO COMPUTER SCIENCE -

### HISTORY & CONTEXT



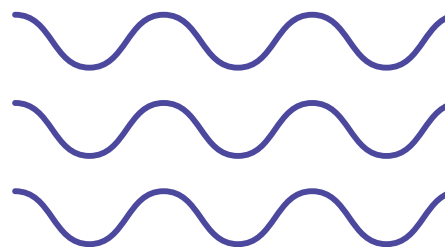
LINNAEUS  
UNIVERSITY  
SUMMER 2024



# TABLE OF CONTENTS

<b>3</b>	what is computer science?	
<b>4</b>	jobs in computer science	
	job examples	4
<b>5</b>	origins of the computer	
<b>8</b>	pioneers of computing	
	Charles Babbage	8
	Ada Lovelace	8
<b>11</b>	computer science - a timeline	
	the 1940-1980s	11
	personal computers	14
	dawn of the internet age	14
	Windows 95	15
	Google	17
	the mobile age	18
	social media	19

<b>22</b>	why is computer science important?	
	dangers & digital ethics	24
	developers responsibilities	26
<b>27</b>	ai - the future	
	big data	28
	Chat GPT & LLMs	30
<b>31</b>	glossary & definitions	



# WHAT IS COMPUTER SCIENCE?

Computer Science is the study of computers - its complex structures and systems - both it's physical components and digital aspects. It includes the algorithms, architecture, and principles behind websites, programs, operating systems, and all the computer software and hardware you've used throughout your life.

Hardware can be categorized as all the parts of a computer you can physically touch with your hands. Examples include hard drives, monitors, motherboards, graphics cards, computer cases, and so on. But mobile phones, computer chips, and flash drives (USBs) equally count as hardware! Anything that needs electricity, that exists physically and performs computation or data processing is considered hardware within computer science. Software, on the other hand, runs on hardware.



Boiled down to its most basic nature and fundamental principle, software is a set of instructions that tells hardware what to do. These commands are written in specific languages that computers understand, and these languages have rules and **syntax** that define how the instructions must be structured and executed. This is what code is - us telling computers what to do and how to do it. Code is put together and then interpreted into software. Software runs on a computer to perform specific tasks and functions, providing executable instructions that the hardware processes to carry out various functions. Each category of software has its own rules, standards, and structures. What they all share is that they enable us to communicate with machines and allow machines (and their components) to communicate with each other.

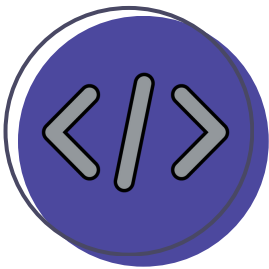
Computer Science is a broad field that encompasses the theory, design, development, and application of computer systems and software. Through understanding and advancing these principles, we continue to improve how we interact with machines, driving progress in countless areas of modern life.

# JOBS IN COMPUTER SCIENCE

As one can imagine, the scope of possible jobs within the field is massive. One can work primarily with hardware, software, or a combination. Some jobs focus on upgrading and fixing software, and some on innovating new products using software.

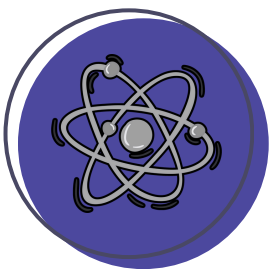
As this is fundamentally a preparatory course for programmers, our focus will be on software.

## JOB EXAMPLES



### ***Software Developer***

is perhaps the most common job associated with computer science. Software developers design software architecture, code, and test computer software or mobile apps. They could be working on various systems, from operating systems and databases to websites, mobile applications, and much more.



### ***Data scientists***

use their knowledge of computer science and statistics to analyze and interpret complex data helping, companies make decisions informed by evidence and metrics. They also streamline data, making it easier to categorize correctly, store efficiently, and retrieve without issue from databases when needed. Data scientists create algorithms, build predictive models, and use visualization tools to understand and present data trends & patterns.



### ***In cybersecurity***

one's responsibilities primarily consist of protecting an organization's data and information from cyber threats. This is accomplished by monitoring an organization's networks for security breaches, investigating violations when they occur, and implementing solutions to secure systems. But it is a wide field, and other tasks could encompass ethical hacking, or testing an organization's security by trying to physically breach an organization's premises.



### ***In interdisciplinary fields***

programmers work closely with specific domains, such as biology, robotics, artificial intelligence, quantum computing, computational neuroscience, digital arts, scientific computing applications, and much more. In bioinformatics, for example, professionals might analyze and interpret complex biological data, or they might develop software tools and algorithms to analyze genomic sequences or to model biological processes.



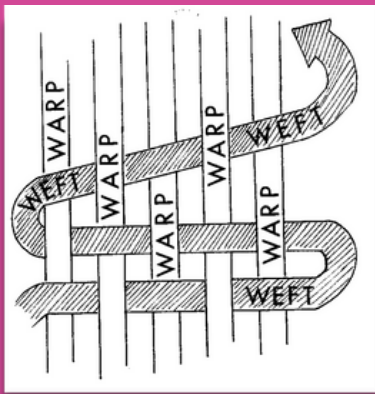
### ***Game programmers***

use programming languages to create video games.

Game programmers most often use a game engine, a **framework** that helps to bridge the gap between art, 3D models, music, sound design, and code. Programmers connect assets such as, for example, character models or menus with some type of functionality, ensuring that players can interact with the game.

# ORIGINS OF THE COMPUTER

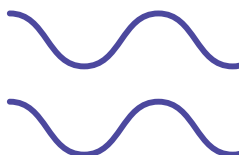
An unexpected precursor to the conception of the computer can be traced back to the loom. In fact, the loom is the ancestor of every computer and smartphone on Earth.



Looms work by interlacing threads to create fabric, with the weaver manually controlling the movement of the threads to form patterns and textures.

In weaving, threads are interlaced at right angles; warp threads run lengthwise and are held taut, while weft threads are woven crosswise through the warp threads.

By raising or lowering specific warp threads, different patterns and textures are created in the fabric.



The Jacquard loom was invented by Joseph Marie Jacquard in 1804.

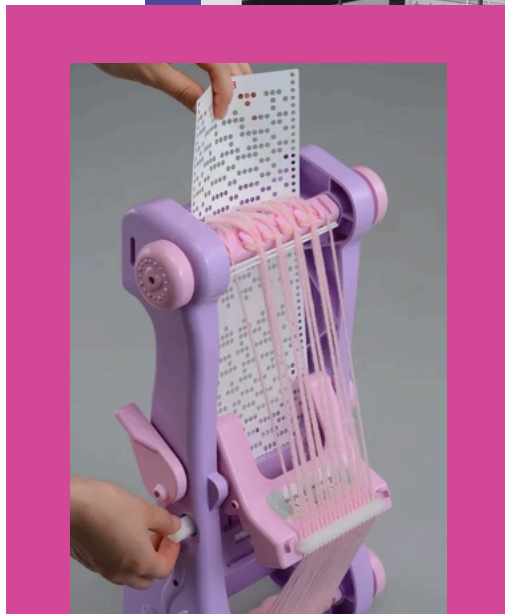
This loom introduced a system of punched cards to control the pattern being woven, where each card represents a row of threads within the fabric.

The presence or absence of holes on each card corresponds directly to each thread in that row: each presence or absence of a hole determines whether a particular thread will be raised or lowered during the weaving process. Threads without a hole stay down because the needle doesn't lift them. So, no hole means the needle doesn't push the thread up, keeping it in place. Thus, the number of holes on a card matches the number of threads in that row, allowing for precise control over the fabric's pattern.



This invention greatly simplifies the process of creating complex patterns and influenced the development of programmable machines - it demonstrated that we can configure machines and set up instructions in a specific way to achieve a desired function.

The same Jacquard loom could use different cards, that varied in the pattern of holes, allowing one single machine to produce a wide range of intricate designs and fabrics.



The ability to change the pattern by changing cards was an important conceptual precursor to the development of computer programming and data entry.

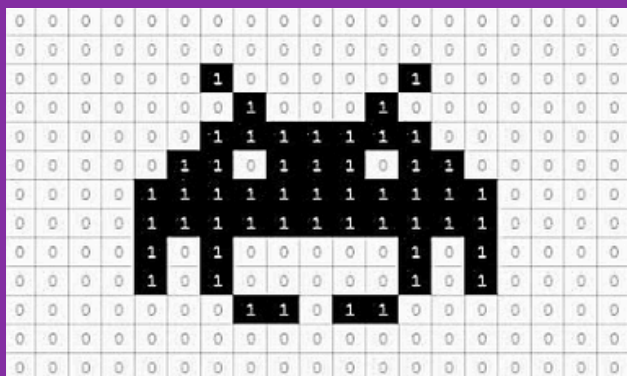
The patterns of the fabrics needed to be represented with these two absolutes. The cards either have a hole, or not, and a thread is either raised or lowered; in either case, there is no intermediate position.

The use of these two oppositional absolutes—raised or lowered threads—allowed for detailed and varied patterns in the fabric.

This **binary** choice of raised or lowered threads mirrors the fundamental principle of **binary code, where information is represented using only two states: 0 and 1.**

Just as the loom uses raised and lowered threads to achieve diverse patterns, **binary code employs two values to represent and control complex computational functions -**

Its dual state enables the creation and manipulation of complex information in computing.



***Much of learning to code is to learn to express oneself in this binary format, and give instructions according to this principle.***

# PIONEERS OF COMPUTING

## CHARLES BABBAGE



Charles Babbage, a very eccentric British inventor and mathematician was inspired by this loom and applied its concept to his field. He wanted to create a machine that could do any math problem you chose for it to do, or "programmed" it to do. He started to work on a mechanical general-purpose calculator-computer in

1837, proposing to call the machine "The Analytical Engine".

You could, similar to the loom, give the Analytical Engine (the hardware) instructions in the form of cards (software) punched with holes (instructions). The holes let pins in, and the absence of holes forced other pins backward, setting off a chain of mechanical calculations.

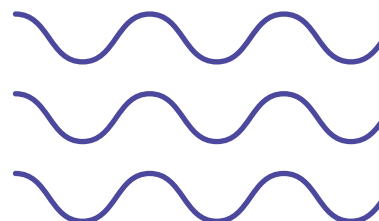
"The Analytical Engine" was a calculator that was meant to do any math problem you would give it. "The Analytical Engine" was designed to perform any mathematical calculation you might give it. While it initially handled only mathematical problems, this concept soon evolved, leading to computers capable of handling tasks beyond simple calculations.

## ADA LOVELACE





Ada Byron, Countess of Lovelace, was the daughter of the renowned Romantic poet Lord Byron.

She was introduced to Babbage's work through their mutual acquaintances and became deeply interested in his machine. She would visit Babbage often, and Babbage was impressed by Lovelace's intellect and analytic skills, even calling her "The Enchantress of Numbers".







During nine months in 1842–43, Lovelace translated the Italian mathematician Luigi Menabrea's article on Babbage's Analytical Engine. With the article, she appended a set of notes, which are around three times longer than the article itself, and included in complete detail, an algorithm for calculating a sequence of Bernoulli numbers using the Analytical Engine. This was the first published **algorithm** ever specifically tailored for implementation on a computer.

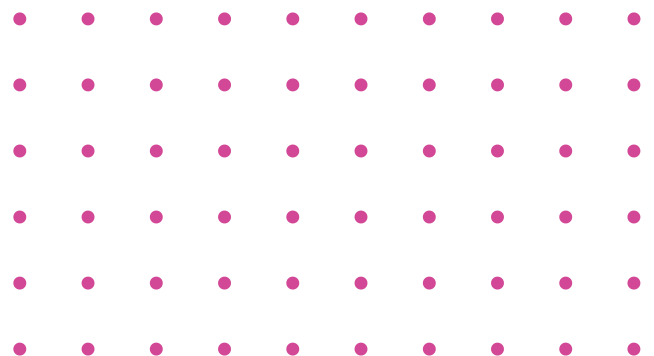
Her notes are considered one of the first instances of computer programming, with her algorithm regarded as the first computer program.

The notes likewise portrayed a visionary explanation of how the engine could be programmed to perform complex calculations and even applications far beyond its initial scope of solving mathematical problems.

***"The bounds of arithmetic were outstepped the moment the idea of applying the cards had occurred."*** -  
Ada Byron, Countess of Lovelace

Ada herself described her approach as "poetical science" and saw herself as an "analyst" and "metaphysician." Looking back, perhaps it is fitting that it took the daughter of a poet to see applications beyond pure calculation.

***...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....***



**According to the historian of computing and Babbage specialist Doron Swade:**

*“Ada saw something that Babbage in some sense failed to see. In Babbage's world, his engines were bound by numbers. What Lovelace saw was that numbers could represent entities other than quantity. Once you had a machine for manipulating numbers, if those numbers represented other things—letters, musical notes—then the machine could manipulate symbols of which number was one instance, according to rules. This fundamental transition from a machine that is merely a number cruncher to one that manipulates symbols according to rules marks the shift from calculation to computation—to general-purpose computation. Looking back from the present high ground of modern computing, if we are sifting through history for that transition, it is clear that Ada made this transition explicit in her 1843 paper.*



**Ada anticipated the implications of modern computing one hundred years before they were realized. Because of her vision, she is often celebrated as the first computer programmer.**



# COMPUTER SCIENCE - A TIMELINE

## THE 1940-1980'S


During the following decades, computer science accelerated astronomically. In the **1940s, large and expensive computers called mainframes were developed and primarily adopted by governments and big organizations.**

The mainframe originated with machines like the Harvard Mark I and ENIAC - these room-sized electromechanical devices were used by a myriad of institutions for extensive data processing needs and to perform complex calculations, which were essential for large-scale operations. However, their high cost and the need for technical expertise made them inaccessible to smaller companies.



Mainframes could easily take up a whole room. Harvard Mark 1, also called the IBM Automatic Sequence Controlled Calculator (ASCC), was the first mainframe computer ever invented. It was over 50 feet wide and 8 feet tall.

Government agencies used mainframes for scientific research, cryptographic analysis and even military operations, most notably with the ENIAC employed by the U.S. Army during World War II.



Universities and scientific organizations also utilized mainframes for advanced research, and even large companies began adopting their use, such as U.S. Steel, which employed them for inventory management and production planning.

Nuclear energy programs utilized mainframes for research and simulations related to nuclear reactions and energy production. The development of the atomic bomb under the Manhattan Project during World War II, for instance, involved significant computational work that used early mainframe computers.

NASA officially started using mainframes in the 1950s, and in fact only just shut down their last mainframe, the IBM Z9 Mainframe, in 2012!

Development continued, and the introduction of vacuum tubes and punched card technology in the 1950s paved the way for mainframes like IBM 701, offering faster processing, lower power consumption, and greater reliability.

**In the 1960s, the second generation of mainframes emerged, made possible by the invention of the transistor.** The transistor is a tiny electronic device that acts as a switch, controlling the flow of a current in electronic circuits. This innovation made computers smaller, faster, and more reliable.

**The third generation System/370 introduced integrated circuits (ICs),** further improving performance and reducing physical size. ICs are compact electronic circuits that combine numerous small components, like transistors, resistors, and capacitors, into a single unit, typically made from a semiconductor material such as silicon. Integrated circuits (ICs) are commonly referred to as computer chips.

These components are interconnected to perform various electronic functions, such as amplification, signal processing, and data storage. This integration allows for the creation of smaller, faster, and more reliable electronic devices compared to those that use separate, individual components. This generation also saw the introduction of virtual memory systems and multi-programming capabilities, enabling concurrent execution of multiple tasks and efficient memory management.

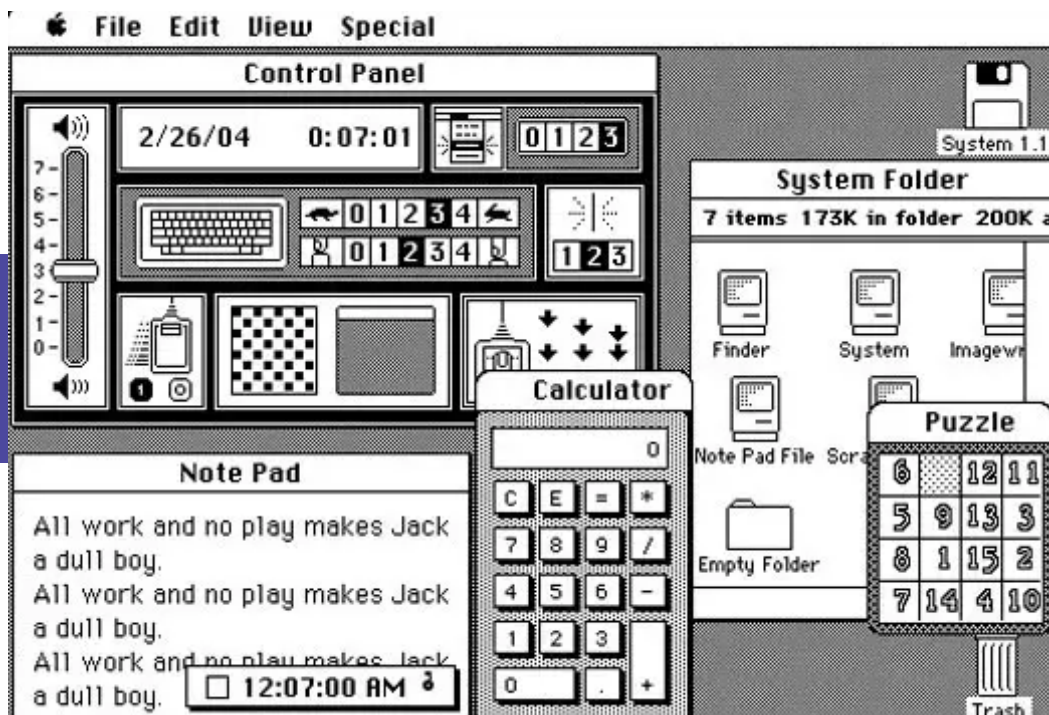


By the 80s, the fourth generation had emerged and brought the widespread use of microprocessors and the development of more powerful **central processing units - CPUs**.

Additionally, mainframes became more user-friendly and accessible, with improved **software interfaces** and networking capabilities, enabling broader adoption in various industries.

**In the late 1970s and early 1980s, the personal computer revolution took off, with the introduction of machines like the Apple II and the IBM PC.** These compact computers were designed for individual use, marking a shift from using one large mainframe for a whole organization. Computing power was becoming increasingly decentralized, allowing individuals and smaller businesses to leverage technology for productivity and innovation without relying solely on mainframes.

**Graphical User Interfaces (GUIs) emerged in the 1980s**, notably with the Apple Macintosh in 1984, and revolutionized computing by introducing visual icons, windows, and menus, making computers more user-friendly and accessible to a wider audience.





# PERSONAL COMPUTERS

**During the 80's the personal computer revolution swept the world.**

As mentioned, in the late 1970s and early 1980s, machines like the Apple II and IBM PC were introduced. These were significantly smaller and cheaper computers, designed for individual use compared to the large, expensive mainframes that had dominated computing thus far.

The emergence of GUIs in the 1980s, such as those on the Apple Macintosh, made computers much easier to use by replacing complex text-based commands with visual icons, windows, and menus.

This innovation drastically lowered the barrier to entry for non-technical users.

At the same time, production techniques improved and the cost of personal computers continued to drop, making them affordable for many households. As the decade progressed, these advancements made computers accessible to “ordinary people” by simplifying their use and reducing costs, allowing individuals to perform everyday tasks like word processing, gaming, and personal productivity at home.

## THE DAWN OF THE INTERNET AGE



The 1990s witnessed several significant advancements in computing technology. **Tim Berners-Lee, while working at CERN, proposed the World Wide Web as a way to access and share information over the Internet using hypertext.** His concept was implemented in 1991, and not long after, introduced beyond the research community to the general public.

**The Internet, a global network of interconnected computers and servers that communicate using standardized protocols, enabled access to the World Wide Web.** This network allowed users to access and navigate web pages—documents written in HTML and linked together using **hyperlinks** through web browsers like Chrome, Firefox, and Safari.



The introduction of the Web transformed information sharing, communication, and e-commerce, enabling a global exchange of data and ideas. During this decade, the Internet became more and more widely available to the public, leading to increased connectivity and the popularization of email as a primary means of electronic communication.

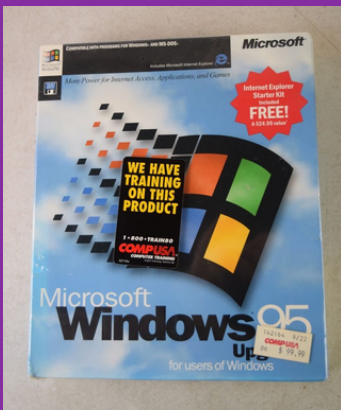
**Internet Service Providers (ISPs)** began offering more affordable and faster access, while browsers like Netscape Navigator and Internet Explorer improved the ease of navigating the Web. By the end of the 1990s, the Internet had become deeply integrated into society, transforming how people accessed information, conducted business, and communicated with one another.

## WINDOWS 95

In 1995 Microsoft released **Windows 95**, a groundbreaking operating system that introduced significant improvements in the **user interface**, capacity for multitasking, and more available plug-and-play capabilities.

*Windows 95 was revolutionary for its introduction of the start menu and taskbar, which provided a user-friendly and intuitive way to navigate applications and files, fundamentally changing how users interacted with their computers.*

*This significantly lowered the learning curve for new computer users, and contributed to a rapid increase in the adoption of personal computers.*

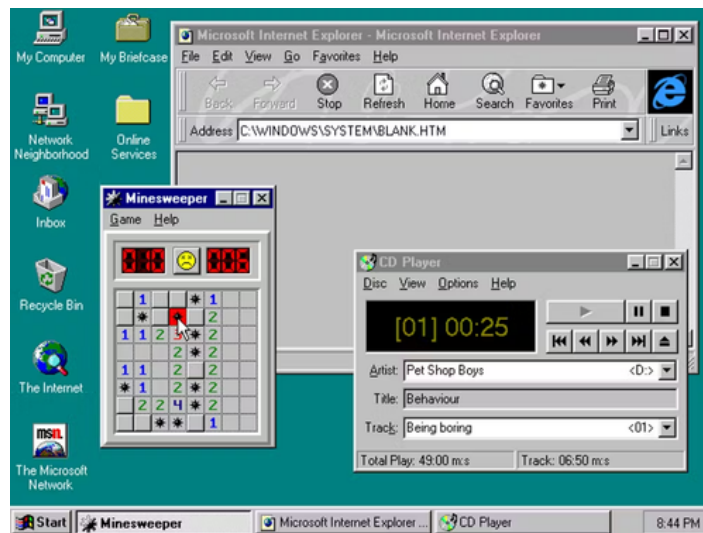


Windows 95 streamlined the user experience and significantly improved the efficiency of operating system functions, popularizing it and setting the standard for future Windows releases.



Components of the Windows 95 GUI became synonymous with computing, and their visual elements even made their way into other media, including print design and television commercials. It was argued that, with the advent of the GUI, engineering had merged with art to create a new medium of the interface.

Today, professionals such as UX designers and digital artists continue to design and develop interfaces, blending functionality with aesthetics to enhance user experience across various digital platforms.



**During the late 90s and continuing into the 2000s, broadband internet became more widely available and affordable,** offering significantly faster speeds compared to dial-up connections.. This facilitated the growth of multimedia content, streaming services, and online gaming, and started changing how people accessed and consumed digital content.

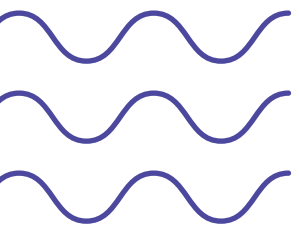
As broadband internet gained traction, traditional media industries underwent significant transformations. Television viewing began to shift from scheduled broadcasts to on-demand streaming platforms, allowing viewers to watch their favorite shows and movies anytime, fundamentally altering television consumption patterns.

The internet's ability to deliver real-time information and multimedia content completely changed the media industry, making digital platforms the dominant sources of news and entertainment.





**Google was launched on September 4, 1998.** It transformed how people access and search for information online, leading to the rise of internet search as a primary tool for navigating the web by providing a highly effective and user-friendly way to find information. With its innovative search algorithm, which prioritized relevance and quality, Google significantly improved search accuracy. Before Google, search engines often delivered less relevant results and were much harder to use. Using Google, users could find the information they needed quickly and efficiently, turning “search” into an essential tool for accessing content, exploring topics, and navigating the increasingly vast internet.



Google’s success also paved the way for the development of various other services and technologies, including email, maps, and cloud computing, profoundly influencing global information access and internet usage.

**Acquired by Google in 2006, YouTube has made video sharing easy for anyone to upload and view content globally. This shift has disrupted traditional media,** moving audiences from TV to online platforms for entertainment and news.

In education, YouTube offers a vast range of tutorials and lectures, democratizing learning and making information more accessible. YouTube and online learning platforms have greatly impacted poorer communities by providing free or low-cost educational resources, helping bridge gaps in access to quality education and improving socio-economic conditions. Hiring practices have adapted to value skills and certifications obtained online, focusing on practical abilities over formal qualifications.

Additionally, the availability of diverse online courses has allowed individuals to gain in-demand skills. For students, these resources have expanded educational possibilities, enabling exploration of new subjects and career paths beyond traditional schooling.

Overall, these technologies have spread far beyond the digital realm, deeply transforming education, entertainment, and everyday life



# THE MOBILE AGE

**Parallel to the Internet age, the 1990s also saw the rise of mobile phones, which became increasingly accessible to the general public.**

The 1990s saw the development of the second generation (2G) of mobile networks, which enabled digital signals and SMS texting.

**In 2000, the introduction of color screens and the first camera phones** marked a significant leap in functionality.

These devices, though primarily used for voice communication, started incorporating basic features such as text messaging and simple games.

The 2000s brought the third generation (3G) networks, allowing for faster internet access and the rise of smartphones.

**In 2007, Apple revolutionized the market, and the world, with the launch of the iPhone.** This set a new standard for mobile phones - and how we interact with them. The iPhone combined a phone, multimedia capabilities, internet access, and a touch-based interface into a single device, giving the world a powerful computer right in our pockets.

The App Store launched in 2008, and its success spurred massive growth in mobile applications, allowing users to customize their devices with a wide array of apps for everything from social media to productivity and gaming.



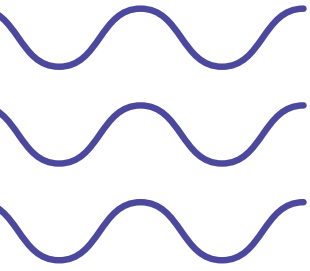


**The Android operating system, launched by Google in 2008, gained significant traction in the 2010s.** Its open-source nature and partnerships with multiple hardware manufacturers led to an explosion of Android devices across various price points, making smartphones more accessible, profoundly impacting technology and society by democratizing mobile computing and app usage.

**With the widespread rollout of 4G LTE networks in the early 2010s,** mobile data speeds were dramatically improved, leading to increased connectivity and a more seamless mobile experience, further integrating smartphones into daily life.

This increased accessibility has become essential in today's digital age, making smartphones into critical tools for communication, work, and daily tasks. The widespread availability of affordable smartphones with high-speed has promoted greater digital inclusion, enabling more people to access information, services, and opportunities previously limited to higher-income users.

**Moreover, the expansion of mobile apps has revolutionized industries, from banking to healthcare, making services more convenient and personalized.** This shift has fundamentally altered how people interact with technology, emphasizing the importance of mobile connectivity in modern life.



**The 2010s also witnessed the emergence of wearable technology, such as smartwatches and fitness trackers,** which integrated with smartphones to provide additional functionality and health tracking. **Smart home devices,** controlled via smartphone apps, became increasingly popular, expanding the role of mobile technology to home automation.

## SOCIAL MEDIA

The popularity of smartphones, played a pivotal role in the emergence and rise of social media. With the advent of affordable, high-speed internet and user-friendly mobile apps, platforms like Facebook, Twitter, Instagram, and Snapchat became integral to the smartphone experience.



These social media platforms altered how people connect, share information, and engage with online communities, leading to an unprecedented level of real-time interaction and content sharing.

Social media's impact on society has been profound. It has reshaped communication by enabling instant, global connectivity and democratizing content creation and distribution.



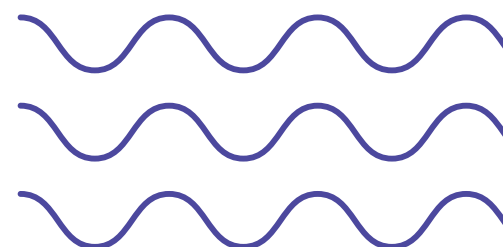
**Media organizations no longer control what is broadcast to the wider public** – with the advent of social media, content creation and distribution have become decentralized, allowing individuals and non-traditional sources to share information and reach audiences directly.

**The integration of social media and mobile technology has also extended to various sectors, including healthcare and public services.** Mobile apps now allow users to book medical appointments, access health records, and receive telehealth consultations, making healthcare more accessible and convenient. Public transportation systems use mobile technology for real-time tracking and ticketing, improving commuter experiences. Similarly, government services have embraced digital platforms to streamline processes and enhance citizen engagement, from online tax filing to social service applications.

**Politically and socially, mobile phones and social media have been catalysts for change. They have facilitated grassroots movements and activism by providing platforms for organizing, mobilizing, and raising awareness on social and political issues.**

This has led to significant political events and shifts, such as the Arab Spring in 2010, where social media played a critical role in disseminating information and rallying support.

Ordinary people can document and share their experiences, with grassroots journalism offering unfiltered accounts of events, particularly in situations where mainstream media may have limited access or face censorship.



# WHY IS COMPUTER SCIENCE? IMPORTANT?

If there is a trend to be seen over the course of computer science history, it is the relentless and lightning-speed integration of technology into daily life, where the boundaries between analog and digital increasingly blur. Early computational machines and mainframes were confined to specialized environments, but as technology advanced, computing became more accessible and embedded into every aspect of our lives.




The rise of personal computers, smartphones, and the internet has made digital tools integral to how we work, communicate, travel, and entertain ourselves... to how we live. The profound impact of this integration on society is evident. Overall, the digital realm has become a fundamental part of modern life, reshaping our civilization.

All of this can be traced back to the loom and its punch cards, where the simple presence or absence of holes laid the groundwork for the binary code that would revolutionize technology.

These binary digits—ones and zeros—form the building blocks of the digital age. The profound impact technology has had on the world is made possible by writing code, which is ultimately distilled into rows and rows of these fundamental binary values.

It is those who write this code who determine what the world will look like—deciding which technologies are developed, how they are used, shaped, and distributed. Coders shape billions of people's lives.





While it is impossible to accurately estimate how much code our world runs on, a rough approximation can be made based on available data and industry trends.

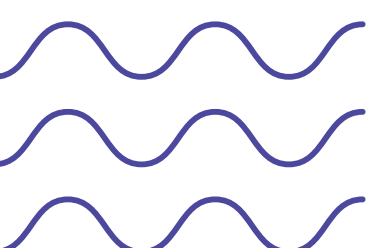
Major operating systems, such as Windows and Linux, have millions of lines of code. **For example, Windows consists of around 50 to 100 million lines of code, and Linux of over 30**

**million lines of code.** Large applications and services, including web browsers (e.g., Google Chrome), office suites (e.g., Microsoft Office), and cloud platforms (e.g., Amazon Web Services), each have millions of lines of code. Millions of apps across platforms like iOS, Android, and the web each contain thousands to millions of lines of code.

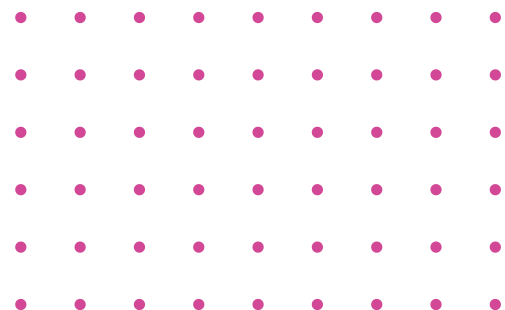
**Aggregating these sources, it's reasonable to estimate that the total number of lines of code in active use globally today could be in the range of several hundred billion lines across various domains.**

As AI is relatively new within the digital landscape, many typical software development projects consist of human-written code, often exceeding 90%. **That means that 90% of all that code was written by programmers.**

With an estimated global population of around 8 billion people and approximately 25 to 30 million professional software developers worldwide, professional developers whose code is actively used represent approximately 0.31% to 0.38% of the world's population.



***Computer science is important because those who write the code - those 0.38% - are the ones shaping the future.***

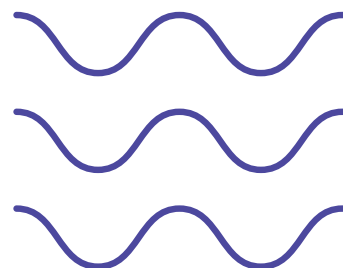
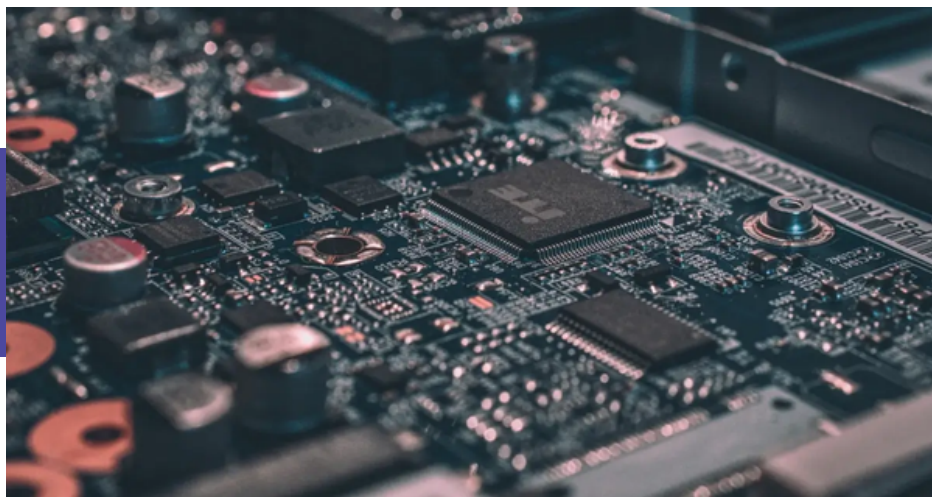


# DANGERS AND DIGITAL ETHICS

As many positive aspects that technology has brought, there are both societal and personal dangers and responsibilities we as users and coders must be aware of.

As digital companies increasingly provide our solutions and services, our reliance on them grows. For example, while YouTube allows anyone to upload videos, it exercises significant control over what content is surfaced and seen by users. This control is managed through algorithms that prioritize certain videos based on factors such as user engagement, relevance, and advertiser preferences. Consequently, the content users are exposed to is often influenced by these algorithms, which can impact everything from the information they receive to the types of media they consume.

This control by digital platforms shapes our information landscape and influences public opinion, often prioritizing sensational or popular content over diverse or nuanced perspectives. As a result, our exposure to information is not just a matter of what we seek out but also what these algorithms deem relevant or profitable, raising questions about content diversity, transparency, and the potential for algorithmic bias.





In times of crisis, should digital services go offline, the lack of robust safeguards and backups can lead to severe consequences. **The dependency on digital infrastructure** and electricity means that disruptions, whether from cybersecurity threats or technical failures, can cripple essential services.

For example, when digital infrastructure fails, it can halt flights, disrupt banking operations, and compromise hospital services, leaving individuals vulnerable and affecting their daily lives.

**Moreover, cyberattacks can result in stolen personal records, leading to significant privacy breaches and identity theft.** The implications of such breaches can be far-reaching, impacting everything from financial security to personal safety.

The critical nature of digital infrastructure underscores the need for comprehensive backup systems and resilience planning to mitigate the impact of service disruptions and ensure that essential services remain functional should something happen.

**Actors can influence what content is accessible to the public, as seen in debates over net neutrality.** For instance, if Netflix were to experience delays loading movies while HBO had instantaneous access, users might switch their subscriptions based on the perceived performance differences. This subtle manipulation of user experience can significantly alter consumer behavior, often without the need for overt regulation.

Such control extends beyond mere choice and can impact broader societal behaviors. By manipulating how digital services perform or which content is prioritized, those in power can exert a regulatory effect that sometimes surpasses traditional legal measures. This underscores how deeply embedded code and digital infrastructure are in our daily lives, **shaping our interactions and decisions in ways that might not be immediately apparent but are profoundly influential.**



In the political arena, control over digital infrastructure and content can have significant effects. **Governments or political groups might censor or block access to information to suppress dissent or limit opposing viewpoints**, such as during political unrest. Social media platforms can manipulate algorithms to prioritize certain content, influencing public perception and political opinions by amplifying specific messages while sidelining others. Additionally, surveillance and data collection can target political advertising. Misinformation and propaganda can spread rapidly, shaping public opinion and swaying elections. Changes in regulations, like those affecting net neutrality, can alter how different services are accessed and prioritized, impacting market dynamics and political outcomes.



## DEVELOPERS RESPONSIBILITIES

Developers bear a unique set of responsibilities and face specific challenges. **They must make a wide range of decisions while creating software, and it's not always possible to foresee every potential scenario.** Unconscious biases can be inadvertently embedded in technology, influencing outcomes in unforeseen ways. Therefore, when teaching programming, it's essential to emphasize the importance of **thinking broadly about various conditions and scenarios, as unexpected interactions and edge cases may arise.** Even so, it is impossible to anticipate every possible scenario. However, by fostering diversity within development teams, we can better mitigate and minimize these issues.

For example - There is a medical condition called hypertrophic cardiomyopathy, which refers to an enlarged heart. Typically diagnosed around the age of 30, individuals with this condition face a higher risk of sudden death before reaching 40. To address this risk, electrophysiologists employ the use of a pacemaker defibrillator, a device that runs on code. However, during pregnancy, when palpitations are a common occurrence affecting about a quarter of all women, the device may misinterpret the palpitations as a dangerous heart rhythm, leading to unnecessary and potentially harmful shocks. If device manufacturers have not extensively considered scenarios involving pregnant individuals, perfectly healthy individuals could receive dangerous electrical currents.

This highlights the importance of having diversity on teams! **By using the lived experiences of a diverse set of individuals**, we have opportunities to consider a wider set of potential variables when creating our software. Ensuring that the future we construct is inclusive and considers the impact on all individuals

## AI - THE FUTURE

Modern AI, particularly in the realm of big data and large language models (LLMs), has evolved significantly from its early roots. The journey began in the mid-20th century with foundational theories in machine learning. However, it was the advent of powerful computing, vast amounts of data, and advanced algorithms in the 21st century that truly accelerated AI's



development. Today, AI technologies harness big data to drive insights across industries, from predicting consumer behavior to optimizing supply chains. Large language models, such as GPT-4, can generate human-like text, understand complex queries, and assist in diverse applications from customer support to creative writing.

These advancements have revolutionized sectors like healthcare, finance, and entertainment, providing unprecedented efficiencies, personalized experiences, and innovative solutions.

However, as AI continues to advance, there are potential risks and challenges. Issues such as algorithmic bias, privacy concerns, and the ethical use of AI need to be addressed to ensure that these technologies are used responsibly and equitably. The future of AI holds great promise, but it must be navigated carefully to mitigate its potential dangers.

## BIG DATA

Big Data AI refers to the use of artificial intelligence algorithms to analyze large volumes of data to extract patterns, and insights, and make informed decisions. While the potential benefits are vast, there is a danger of built-in biases within these systems affecting outcomes. To understand how biases work in Big Data AI, it's important to recognize that these systems **learn from historical data patterns**. If the data contains biases, the AI algorithms can unintentionally amplify and perpetuate them. Biases can also be unintentionally incorporated into algorithms due to various factors like biased data sources, biased data labeling, or biased decision-making during the development process (all of which are managed by biased humans). This can perpetuate discriminatory outcomes, affecting individuals and large-scale communities. **Biased algorithms can lead to unequal treatment**, perpetuate stereotypes, and reinforce societal inequalities.



For example, early facial recognition systems often showed significant bias against people with darker skin tones. These systems were trained on datasets that predominantly featured lighter-skinned individuals, leading to lower accuracy in identifying and verifying people from marginalized groups.





AI-driven hiring tools have been found to exhibit biases that can disadvantage women and minority candidates. If these algorithms are trained on historical hiring data where certain groups were underrepresented or unfairly treated, **they can perpetuate and even exacerbate existing biases in the recruitment process.**

In healthcare, AI systems used for diagnosing diseases or predicting patient outcomes can also be biased. For example, **if a dataset used to train a diagnostic tool is not sufficiently diverse, the tool may perform poorly for underrepresented groups.** For example, if a dataset used to train an AI for detecting heart disease primarily consists of data from male patients, the tool may not recognize symptoms that are more common in female patients. This can lead to misdiagnosis or underdiagnosis in women, resulting in disparities in treatment and healthcare quality.

**Algorithms used in predictive policing and sentencing have shown biases against racial and ethnic minorities.**




crime data, which can reflect entrenched societal biases and systemic inequalities, especially without historical context.

For instance, if historical data disproportionately represents certain racial or ethnic groups due to over-policing or socioeconomic factors,

AI algorithms trained on this data may reinforce and magnify these biases.

This can lead to **false positives in profiling**, where marginalized groups are unfairly targeted based on skewed historical patterns rather than actual risk. Additionally, the failure to account for historical context and socioeconomic status can result in these algorithms reinforcing existing disparities, leading to disproportionately harsher sentences for already marginalized communities.

**This highlights the need for diverse and representative data sets** and ongoing monitoring and evaluation of AI systems to mitigate biases, as well as the importance of prioritizing diverse development teams. Diversity brings together a range of perspectives, experiences, and cultural understanding, which can help identify and address potential biases in data collection, algorithm design, and decision-making processes.





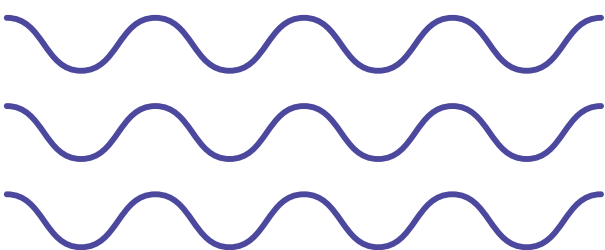
# CHAT GPT AND LLMs

Large Language Models (LLMs) like GPT-4 obtain their information from extensive datasets that include text from books, articles, websites, and various other publicly available sources. This diverse range helps the model learn about different topics and language styles.

However, **if the dataset includes inaccuracies—such as outdated information, unverified facts, or biased viewpoints—the model learns from this flawed data.** During training, the LLM processes this text by tokenizing it into manageable units and then uses pattern recognition to understand relationships, syntax, and semantics. The model adjusts its internal parameters to minimize errors, enhancing its ability to generate accurate and contextually appropriate responses.

When interacting with a user, the LLM first tokenizes the input text and analyzes its context and intent. Using its learned patterns, **the model predicts the most relevant next word or phrase**, generating a response that aligns with the input. This process relies on probabilities derived from its training data rather than genuine comprehension.

However, this workflow introduces several potential risks. LLMs can inherit and perpetuate biases present in their training data, leading to discriminatory or prejudiced responses that affect marginalized groups. Additionally, the model might unintentionally spread misinformation or **amplify false narratives due to its reliance on plausible-sounding but potentially inaccurate content.** The lack of true understanding means that LLMs can generate superficial or incorrect answers, especially on nuanced topics. There are also concerns about privacy and security, particularly if training data includes sensitive information or if the technology is exploited for malicious purposes.





# GLOSSARY & DEFINITIONS

## SYNTAX

the set of rules that defines the structure of statements in a programming language. It dictates how code must be written and arranged to be correctly understood and executed by a compiler or interpreter.

## FRAMEWORK

a pre-built, reusable set of libraries or tools designed to help developers. It provides a structured foundation and guidelines for development.

## BINARY

a base-2 numeral system used in computing and digital electronics. It represents data using only two symbols: 0 and 1.

## ALGORITHM

a step-by-step procedure or formula for solving a problem or performing a task.

## CPU

the "brain" of the computer. The hardware component responsible for executing instructions and processing data

## SOFTWARE INTERFACE

a point of interaction between different software components or between the user and the software.

## UI (USER INTERFACE)

the means by which a user interacts with a computer or software application.

## GUI (GRAPHICAL USER INTERFACE)

a type of user interface that uses visual elements such as windows, icons, and buttons to interact with the user. Unlike text-based interfaces, GUIs allow users to perform tasks through graphical representations rather than commands.

## INTERFACE

a shared boundary or point of interaction between different systems, devices, or components.

## UI DIFFERENCES

**Interface** Broad term for any interaction point, not limited to software or user interactions.

**Software Interface** Specific to interactions between software systems or between software and users, focusing on how software communicates and integrates.

**UI** Refers to the overall design and interactive elements of software that allow user interaction, including all types of interfaces.

**GUI** A subset of UI that specifically involves graphical elements, making it visually oriented and often more user-friendly than text-based interfaces.



## **WORLD WIDE WEB**

a system of interlinked hypertext documents and multimedia content accessed via the Internet. It uses web browsers to retrieve and display information from servers

## **INTERNET**

a global network of interconnected computers that communicate using standard protocols (such as TCP/IP).

## **HYPERTEXT**

a method of organizing and linking text in a way that allows users to navigate between related pieces of information through clickable links.

## **HYPERLINKS**

clickable elements in a web page or document that direct users to another location, such as a different web page, section of the same page, or external resource.

## **ISP (INTERNET SERVICE PROVIDER)**

a company or organization that provides individuals and businesses with access to the Internet. ISPs offer various services, including broadband, dial-up, and fiber-optic connections.