

U O R T E X



Estakisha's medium articles - I

The Withering Console Tree

Culture arises and unfolds in and as play - Johan Huizinga

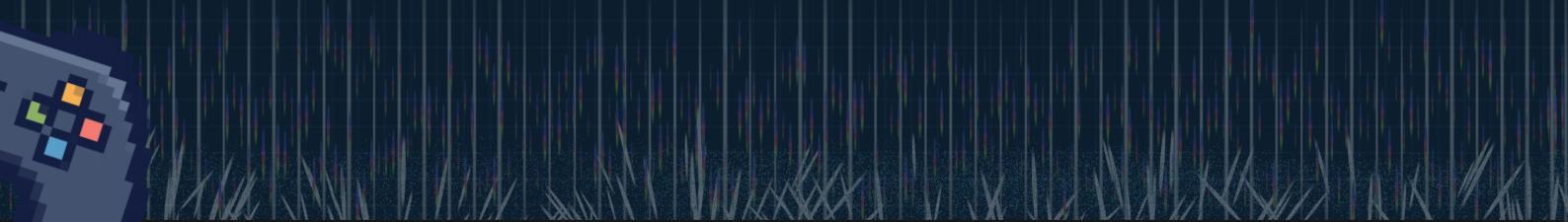
Games have been a part of humanity, arguably being the oldest form of social interactions. They captured the age and culture it grew in, passing down memories and traditions from one generation to the next.

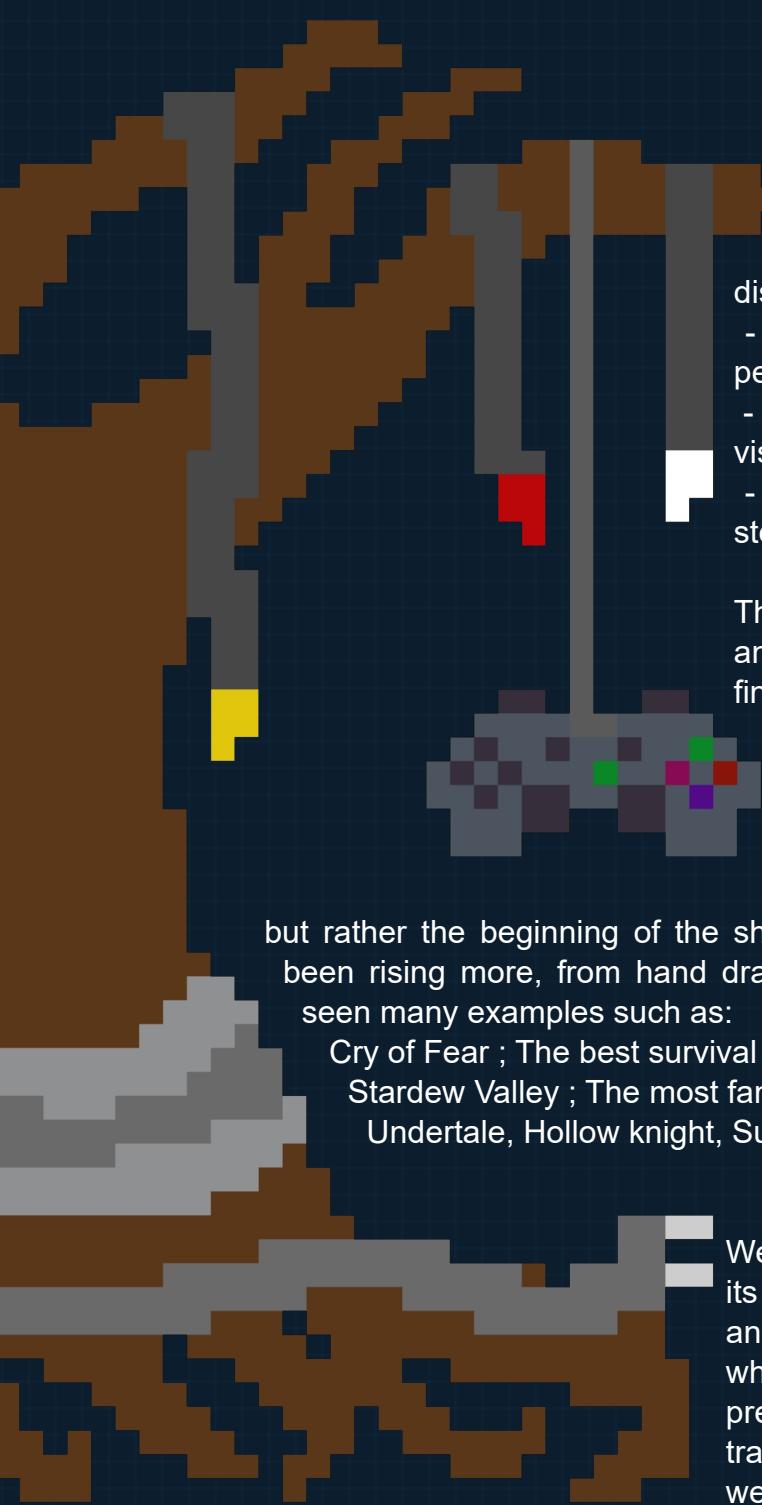
One form of how games changed with time, is the emerging of video games that is traced back to 1950's, at first it was simple mini-games but with time games became more than just something to play, this is what made video games beloved -surpassing even board games for many- was the depth of experience they could offer. Some held carefully crafted narratives and lessons to learn. Other games painted vivid worlds and memorable characters, enabling players not just to play, but to feel the game, whether it was the emotional gravity of "Final Fantasy" or the haunting atmosphere of "Silent Hill", nonetheless, a question asked it self:

What happened to the creativity and uniqueness in the gaming industry?

Let us look into some new recent games, like "Minds Eye" or "Redfall" , so where did we go wrong? The answer is simple, companies fear creative risks and this is exactly why a lot of games are a copy-paste of each other, from the narrative down to the game mechanics, chasing trends rather than adopting and developing their own ideas, AAA games are at a decline.

AAA (or Triple-A) games are high-budget, high-profile video games typically developed by major studios and publishers





So let's face it, in the recent years most companies produced games that are about:

- Mindless farming and grinding designed to waste time
- Predatory gacha system exploiting addiction disguised as "players luck"
- Shallow cloned characters — whether in looks or personality wise
- Market safe design philosophies that erase artistic vision
- Long scripts of nonsense instead of a meaningful story

These games used to challenge players mentally and emotionally, now they are challenging them financially, giving a slow death to the known AAA gaming industry.

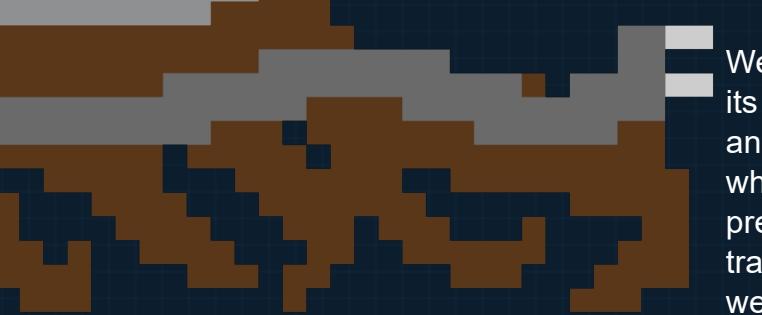
The new hope

However with all that is going on it isn't the end but rather the beginning of the shift, as AAA games are falling, indie games have been rising more, from hand drawn pixels to emotional storytelling and we have seen many examples such as:

Cry of Fear ; The best survival horror game to exist

Stardew Valley ; The most famous farming game

Undertale, Hollow knight, Subnautica are extra honorable mentions



We aren't witnessing the death of gaming only, but its rebirth as well, a future where solo developers and small teams can surpass big corporations, where innovation, emotion, and honesty take precedence over photo-realistic graphics and micro-transactions, from someone that still remembers why we play in the first place.

When Worlds Collide

The Cross Play Revolution



Every age has its storytelling form, and video gaming is a huge part of our culture.

- Andy Serkis

Games have and still do play a huge role in our lives, shaping our logic and affecting our imagination, and these games range from multiplayer online to offline singular play, a lot of us hold special memories with at least one game!.

Today when we talk about famous games, we talk about the biggest games out there, most of them are multiplayer cross platform , but we are here to talk about another cross platform, something that you'd rarely think is possible to happen but it did !.

Cross platform is the concept of playing a single game on multiple devices without losing content and such any game going toward that path has a higher chance of success than others, the logic that goes behind it is quite complicated, so imagine if two games had the ability to do that ?

Minecraft

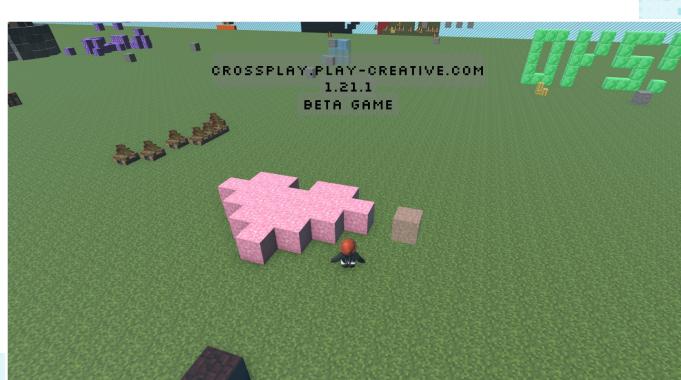
If one of the most famous games for its unique base - is Minecraft , the game it self doesn't have much, after all its just a simple survival game, but what you can make out of it is the difference, from creating in-real-life like structures with simple cubes to building a functional mini-computer nothing stops the creative Minecraft players.

Roblox

Talking about another cross platform game, Roblox , known for being a game engine that allows you to publish your games for free or paid, and if you are just a player, you'll see a variety of games in different genres ranging from cooking, survival or role play you name it! these games can be as creative as it can get or a clone of another famous game, but one thing to note is that this player base has quite the determination to make games more accessible.

So what if I told you, that these fans made a cross-game between these two games? You'd think its impossible since Roblox is a lua based game and Minecraft is java but this fan base is crazy and creative enough to make it happen and they did! and not only one, but THREE projects!

First on the list is Block Game cross play it isn't that visually appealing but it definitely has its unique charm to it, another thing to note is that building in it isn't for free for Roblox players.

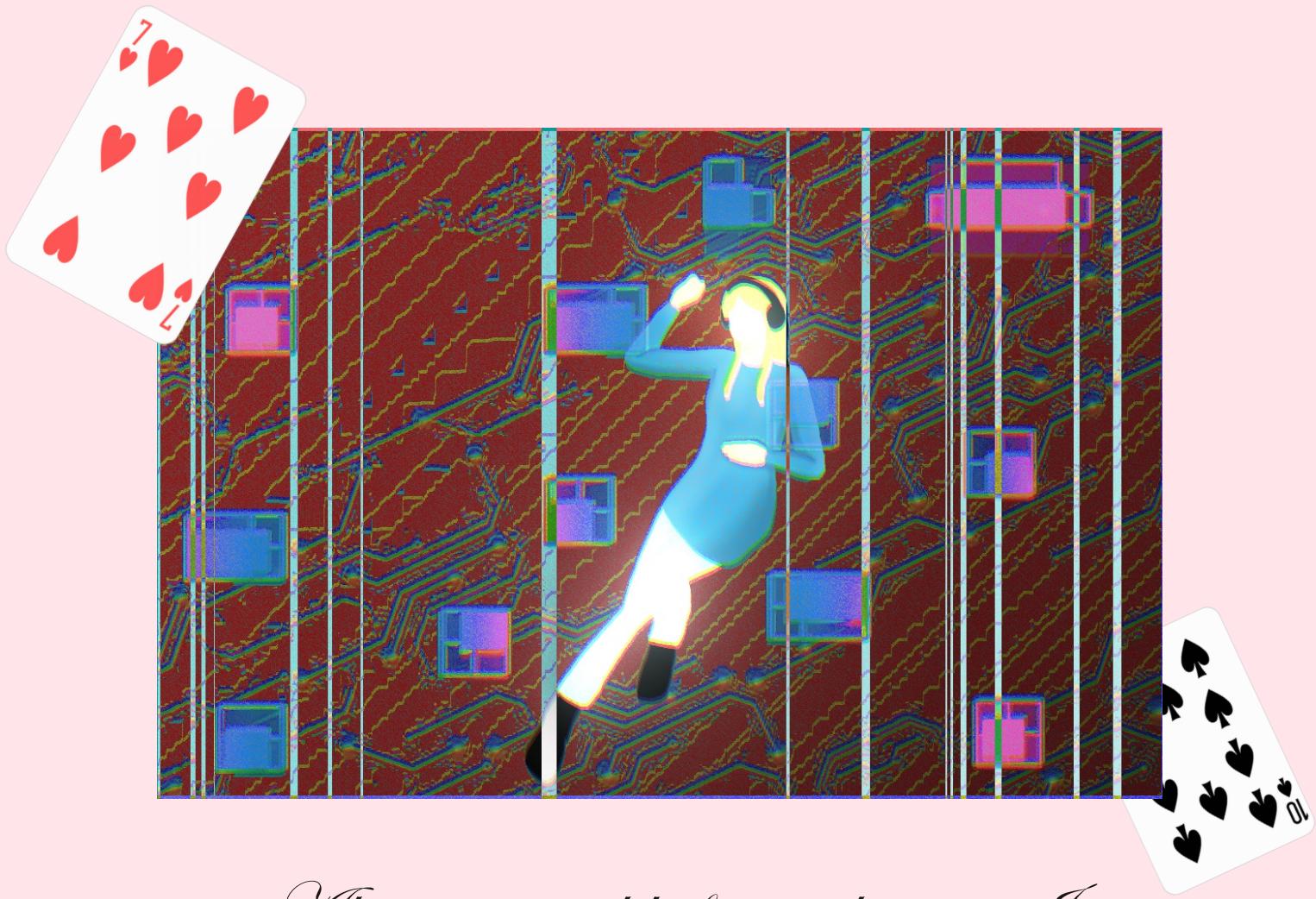


Second on the list the Crossplay Project where Roblox players can build and fly for free, however you wont be able to see Minecraft players or interact with them sadly.



And finally — the best for the last — Mineplay, where Roblox players can build in it for free, and see Minecraft players too, as well as it is more visually appealing than the two before.

Looking at these projects that hold a lot of complexity, not only is it a new step to re-defining what cross play is but also an art work of how it was possible to make it happen with the networking and code tweaking that went behind the screen, this concept came to life by the efforts of a community, exact names aren't known for sure but definitely someone who has the time and energy to make this happen. Nether less its crazy how two different games can collide into a chaotic masterpiece! just a proof to how creative the human mind can be.



Alice in a world of virtualization I

Once upon a time, in the era of codes and technology, there was Alice, in her blue hoodie reading a book on how to code, but her sudden interest in this world didn't spark out of nowhere. Alice's passion was ignited by a legendary developer Mr.Mad Coder known for releasing unique and creative applications every few days. Inspired by his work, Alice set out to create her very first app.

From article to article, book to book, Alice learned how to code, but one thing that stood in her way, how can she test her applications? and she desperately needed a safe space to test her apps, and from here on, Alice finds herself standing before a glowing gateway of virtualization.





The Hare's container

The first one to help Alice, her old friend The White Hare, where he saw Alice looking for a book in the library.

"Are you looking for a book to deploy your app, Alice?" he asked, knowing exactly her purpose. Alice replied with a nod, knowing and trusting that The White Hare will help her.

"Look into using a container Alice".

Alice's eyes widened with wonder, "what is that?" she asked curiously.

"The container is a type of virtualization technology, virtualization is the technology that you can use to create virtual representations of different operating systems allowing a computer to share its hardware resources with multiple digitally separated environments, a container, as I mentioned previously, is a virtualization technology, however here instead of downloading a full operating system, you download only the necessary parts to help you run your applications, it's more like a box that only holds the things your app needs to run, making it lightweight and portable".

Alice wondered and asked, "what if something goes wrong?".

"Well, that is the risk with using a container, as you see, it isn't isolated as some people think. It shares the same resources as the host (which is the device that runs it) but even though it can pose a risk, if you configure it correctly and follow best practices the risk is minimum".

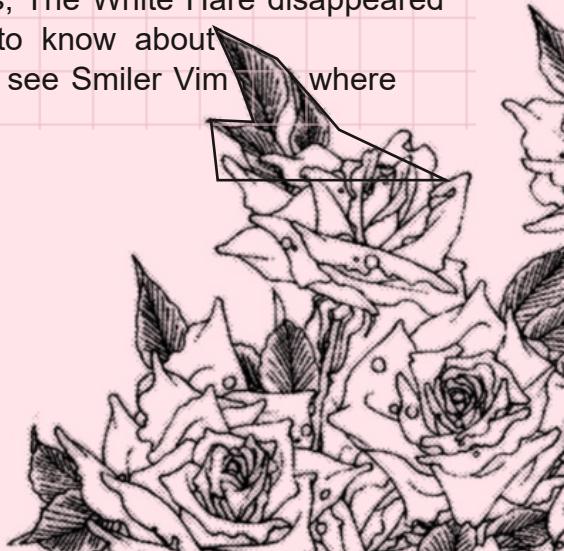
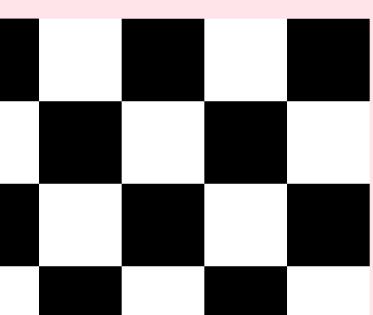
Alice sighed, "Well, that seems like a lot of effort to be done, aside from that how easy is it to navigate in a container?".

The White Hare laughed "Well Alice, nothing is easy if you want something secure and efficient at the same time". and then he continued "If we want to talk on how easy is it to use it, well then it has quite the learning curve, you won't be able to use it in a graphic user interface, but rather as a command line interpreter"

Alice thought and replied, "I will be giving it a thought White Hare, thank you a lot for your insight!".

"Welcome Alice! mind you it's not the only way, but it's the first that crossed my mind".

And before Alice had the chance to ask any more questions, The White Hare disappeared behind the vast library shelves, leaving Alice determined to know about virtualization, just as she turned around, she was surprised to see Smiler Vim where her journey in learning more only began.



THE SUMMER OF 2025

From open iced tea bucket, to verification laws to “protect” children (funny how the EU plans to contradict its own privacy laws), so what didn't happen this summer *yet*? of course who else than Google with their new plans that will apply on everyone by 2027, the nail in the coffin as the summer ended.

As we have seen the recent news, cookies and trackers aren't enough, now we are seeing laws and corporate policies aimed to further kill the privacy of millions by asking for an ID as a form of online validation, and of course Google, known with their love and obsession for hoarding data, joined the wave by future plans to force Android apps to have verified developers, meaning they are turning android to a closed surveilled ecosystem.

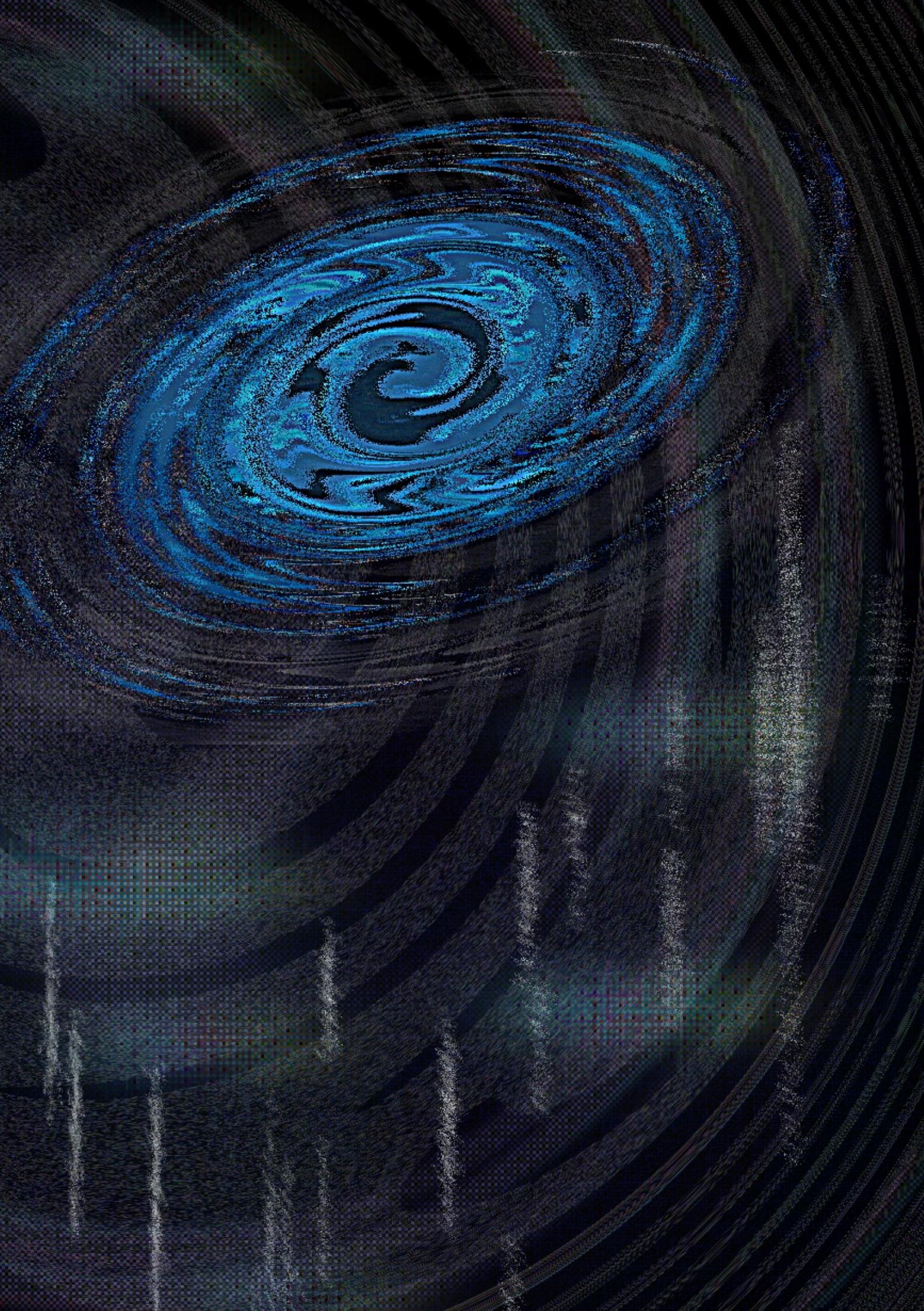
For context, Android is owned by Google since 2005, it was built using Linux kernel to be open, free, and user driven. However as we have seen for the past decades, even though Android has been being shipped with default google apps (GAPPS), that threaten user privacy, it was still favored for the customization options and freedom, but as of recent news, this will go away, harming the independent developers who make their own tools, customization and creative ideas that are outside the reach of google play such as f-droid, GitHub hosted apks and itch.io games.

From a general view to their future plans, this might be seen as good news; no more malware and less shady software, but this misses the point. Freedom always carries risks, and sanitizing the internet under the guise of safety always means surrendering control. And with these changes, Android moves further from its open-source roots, closer to becoming another closed, surveilled walled garden.

If this is what digital safety looks like, privacy will not survive another summer.







The human capacity to find inspiration in the simplicity of the world around us is unending, reflected in the diverse arts and creations that define our civilization. From era to era, we witness the beauty of science birthed from art. This technological age is no different its roots were woven by the hand of art, mother of knowledge.

Holding our phones, scrolling the day away, does the question of how we made this ever pop up? what inspired the pixels on the screens, and the bits that make the machine? where else but in the industrial regime.

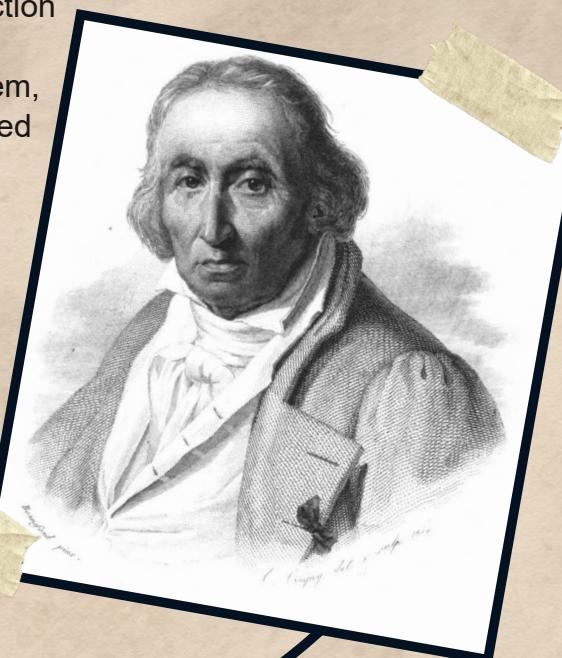
WEAVING THE WIRES

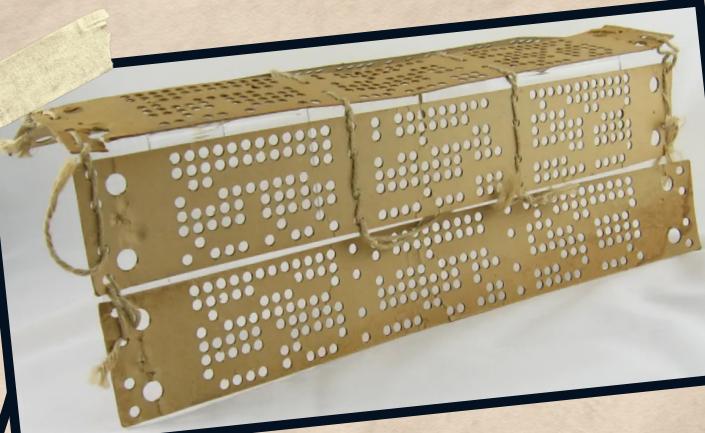
THE LOOM

Lets go back in time to the 1800's, where damasks and textiles were in high demands for their unique looks this of course came with a high price and exhausting work, sparking the interests of Joseph Marie Jacquard, a French weaver and inventor of the Jacquard loom in 1804.

Jacquard loom is a mechanical loom that automates the weaving of complex patterns in fabric. Using punched cards to control individual warp threads, it can produce intricate designs like brocades, damasks, and lace like motifs without manual selection of threads.

This innovation replaced the labor-intensive "drawboy" system, greatly increasing efficiency and consistency in patterned weaving.





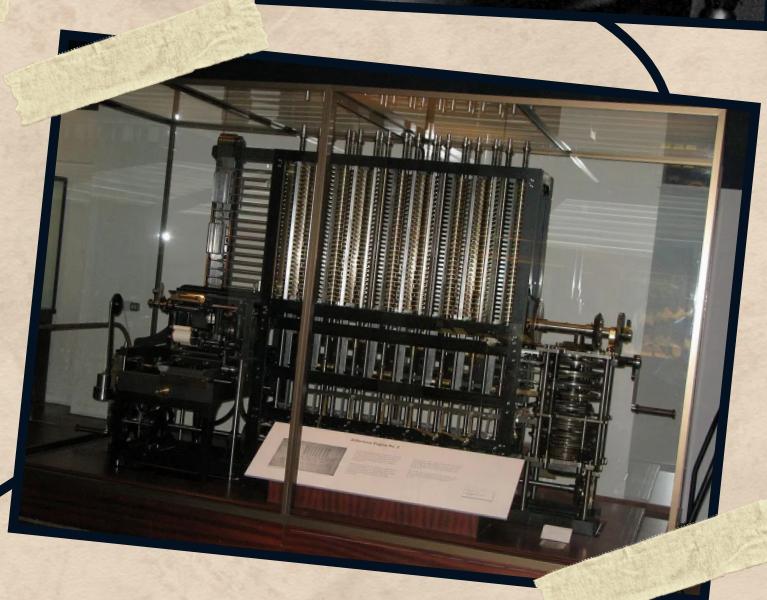
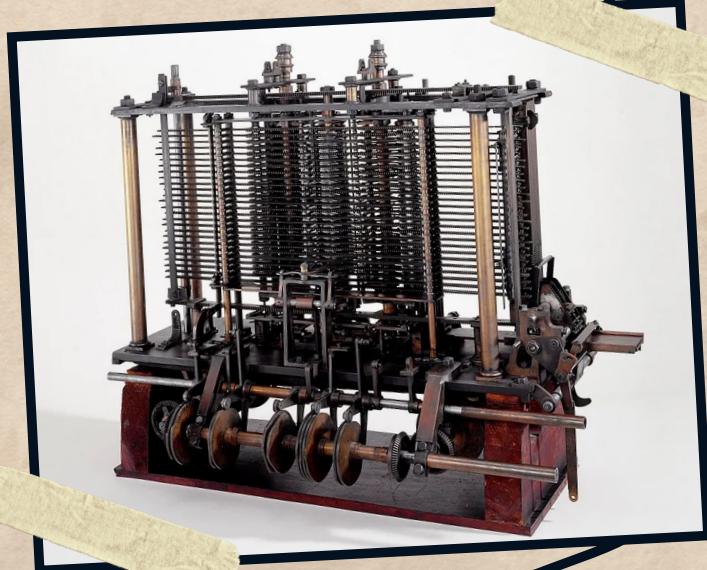
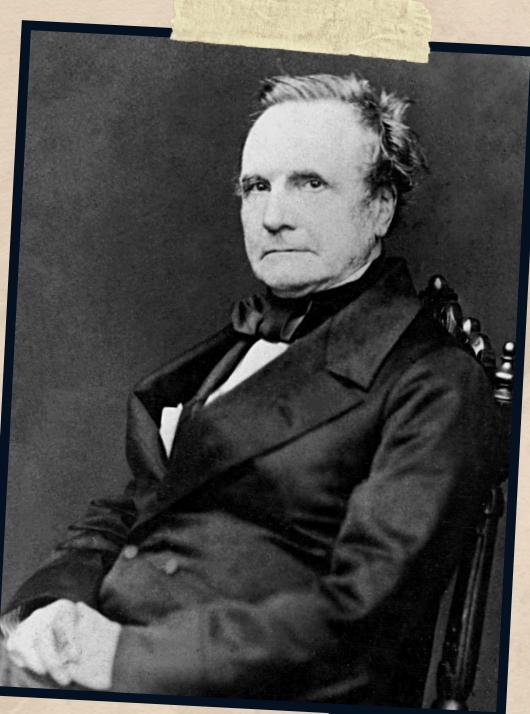
What really makes this machine interesting is the punch cards, where each card is a rectangular piece of stiff paper with holes punched in specific positions, this makes each row of holes corresponds to one line of the woven pattern, and each column corresponds to a warp thread.

When the card passes through the loom, metal pins detect the holes, if there is a hole, the corresponding warp thread is lifted, allowing the weft to pass beneath it, and if there's no hole, the thread stays down. By chaining many cards together, the loom can follow a sequence of instructions, creating a complex pattern automatically. This proved that complex, repetitive processes could be automated through external instructions.

CALCULATED STRINGS

Thanks to Charles Babbage back in 1820's — 1840's , it's when the first conceptual design of a modern computer emerged. Starting with the Difference Engine (1822) which is a mechanical calculator that could perform addition and subtraction automatically by using gears and levers, this helped in avoiding human errors.

Later on, the Analytical Engine (1837) appeared, where Charles Babbage expanded on the Difference Engine by taking inspirations from the Jacquard loom to create a general purpose mechanical computer, where it included: a punched cards to store programs, a mill (like a CPU) for calculations, and a store (like memory) for numbers.



BEYOND THE NUMBERS

Ada Lovelace (1815–1852), was a visionary mathematician she held an unexplored vision of general purpose computing, by studying Babbage's design of the Analytical Engine she recognized that a machine could go beyond calculation.

In her detailed annotations, known as the "Notes"(A-G), in her Note G she described how the Analytical Engine could calculate Bernoulli numbers, a step by step sequence of instructions. This is considered the first algorithm ever written for a machine, making her the first computer programmer.

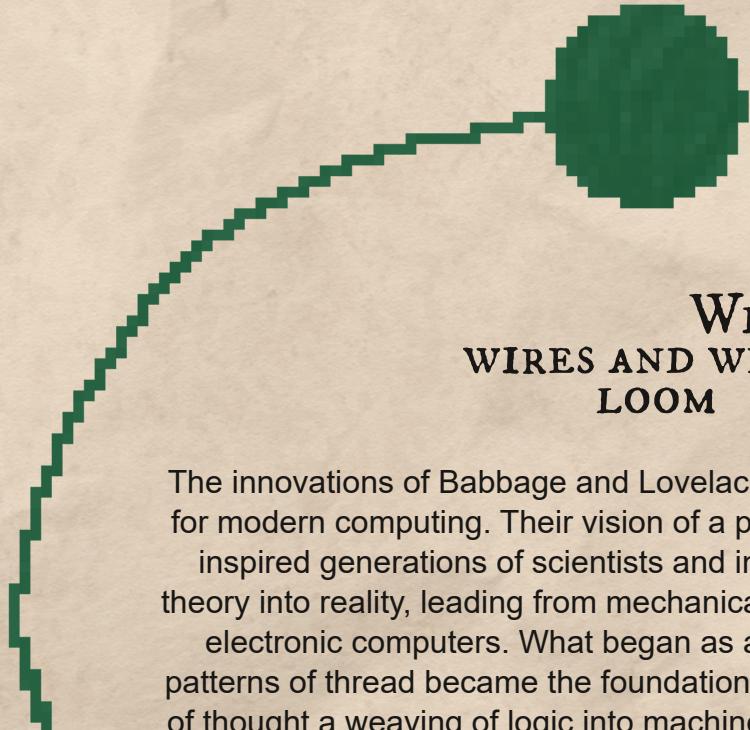
In addition as she stated in Note A Lovelace foresaw that such a machine might one day compose music, art, or design, provided those processes could be expressed mathematically a vision of creative computation that would not be realized until a century later.

Number of Operation.	Nature of Operations.	Variables acted upon.	Variables receiving results.	Indication of change in the value on any Variable.	Statement of Results.	Data.	Working Variables.										Result Variables.				
							V_1	V_2	V_3	oV_4	oV_5	oV_6	oV_7	oV_8	oV_{10}	oV_{11}	oV_{12}	oV_{13}	oV_{14}	B_1 in a decimal fraction.	B_2 in a decimal fraction.
1	\times	$V_2 \times V_3$	IV_4, IV_5, IV_6	$\{IV_5 = IV_2\}$ $\{IV_6 = IV_3\}$	$= 2n$	1	2	n	2n	2n	2n										
2	-	$IV_4 - IV_1$	$2V_4$	$\{IV_4 = 2V_4\}$	$= 2n - 1$	1	2n - 1												
3	+	$IV_1 + IV_2$	$2V_3$	$\{IV_1 = 2V_3\}$ $\{IV_2 = 2V_3\}$	$= 2n + 1$	1	2n + 1												
4	+	$2V_2 + 2V_4$	IV_{11}	$\{2V_2 = oV_{11}\}$ $\{2V_4 = oV_{11}\}$	$= 2n - 1$	1	2n + 1											
5	+	$IV_{11} + 2V_2$	$2V_{11}$	$\{IV_{11} = 2V_{11}\}$ $\{2V_2 = 2V_{11}\}$	$= \frac{1}{2} \cdot 2n - 1$...	2	$\frac{2n - 1}{2n + 1}$					
6	-	$oV_{12} - 2V_{11}$	IV_{12}	$\{oV_{12} = oV_{11}\}$ $\{2V_{11} = oV_{12}\}$	$= \frac{1}{2} \cdot 2n - 1 = A_0$	$\frac{1}{2} \cdot 2n - 1$	$\frac{2n - 1}{2n + 1}$				
7	-	$IV_2 - IV_1$	IV_{10}	$\{IV_2 = IV_3\}$ $\{IV_1 = IV_1\}$	$= n - 1 (= 3)$	1	...	n	0	...	$-\frac{1}{2} \cdot 2n - 1 = A_0$			
8	+	$V_2 + V_5$	V_2	$\{IV_2 = IV_2\}$ $\{IV_5 = IV_2\}$	$= 2 + 0 = 2$...	2	n - 1	...				
9	+	$IV_6 + IV_7$	$2V_{11}$	$\{IV_6 = 2V_{11}\}$ $\{IV_7 = 2V_{11}\}$	$= \frac{2}{2} = A_1$	2	...				
10	\times	$IV_{21} \times IV_{11}$	IV_{12}	$\{IV_{21} = IV_{12}\}$ $\{IV_{11} = 2V_{12}\}$	$= B_1 \cdot \frac{2}{2} = B_1 A_1$	$\frac{2}{2} = A_1$					
11	+	$IV_{12} + IV_{13}$	$2V_{12}$	$\{IV_{12} = oV_{13}\}$ $\{IV_{13} = 2V_{12}\}$	$= -\frac{1}{2} \cdot 2n - 1 + B_1 \cdot \frac{2}{2}$	$\frac{2}{2} = A_1$	$B_1, \frac{2}{2} = B_1 A_1$				
12	-	$IV_{10} - IV_1$	IV_{10}	$\{IV_{10} = 2V_{10}\}$ $\{IV_1 = IV_1\}$	$= n - 2 (= 2)$	1	0	$\left\{ -\frac{1}{2} \cdot 2n - 1 + B_1, \frac{2}{2} \right\}$			
13	-	$IV_6 - IV_1$	IV_6	$\{IV_6 = 2V_6\}$ $\{IV_1 = IV_1\}$	$= 2n - 1$	1	n - 2	...				
14	+	$IV_1 + IV_2$	IV_7	$\{IV_1 = IV_7\}$ $\{IV_2 = IV_7\}$	$= 2 + 1 = 3$	1	2n - 1											
15	+	$2V_6 + 2V_2$	IV_8	$\{2V_6 = 2V_8\}$ $\{2V_2 = 2V_8\}$	$= \frac{2n - 1}{3}$	1	3	...									
16	\times	$IV_8 \times IV_{11}$	IV_{11}	$\{IV_8 = oV_{11}\}$ $\{IV_{11} = 2V_8\}$	$= \frac{2}{2} \cdot 2n - 1$	2n - 1	3	$\frac{2n - 1}{3}$										
17	-	$IV_6 - IV_1$	IV_6	$\{IV_6 = 2V_6\}$ $\{IV_1 = IV_6\}$	$= \frac{2}{2} \cdot 2n - 3$	0	$\frac{2}{2} \cdot 2n - 3$					
18	+	$IV_1 + 2V_7$	IV_7	$\{IV_1 = IV_7\}$ $\{2V_7 = IV_7\}$	$= 3 + 1 = 4$	1	2n - 2											
19	+	$IV_6 + IV_2$	IV_9	$\{IV_6 = 2V_9\}$ $\{IV_2 = 2V_9\}$	$= \frac{2n - 2}{4}$	1	4										
20	\times	$IV_9 \times IV_{11}$	IV_{11}	$\{IV_9 = oV_{11}\}$ $\{IV_{11} = 2V_9\}$	$= \frac{2}{2} \cdot 2n - 1, \frac{2n - 2}{4} = A_3$	2n - 2	4	$\frac{2n - 2}{4}$...	$\left\{ \frac{2n - 2}{2}, \frac{2n - 1}{3}, \frac{2n - 2}{3} \right\}$								
21	\times	$IV_{22} \times IV_{11}$	IV_{12}	$\{IV_{22} = 2V_{12}\}$ $\{IV_{11} = 2V_{12}\}$	$= B_3, \frac{2}{2} \cdot 2n - 2 = B_3 A_3$	0	0						
22	+	$2V_{12} + 2V_{11}$	IV_{12}	$\{2V_{12} = oV_{12}\}$ $\{2V_{11} = 3V_{12}\}$	$= A_0 + B_1 A_1 + B_2 A_2$	0						
23	-	$IV_{10} - IV_1$	IV_{10}	$\{IV_{10} = 3V_{10}\}$ $\{IV_1 = IV_1\}$	$= n - 3 (= 1)$	1		$\{A_2 + B_1 A_1 + B_2 A_2\}$				
24	+	$4V_{13} + oV_{24}$	V_{24}	$\{4V_{13} = oV_{13}\}$ $\{oV_{24} = 2V_{24}\}$	$= B_7$						
25	+	$IV_1 + IV_2$	IV_3	$\{IV_1 = IV_3\}$ $\{IV_2 = oV_6\}$ $\{oV_6 = oV_7\}$ $\{oV_7 = IV_3\}$	$= n + 1 = 4 + 1 = 5$	1	...	n + 1	0	0						

Here follows a repetition of Operations thirteen to twenty-three.

by a Variable-card.

by a Variable card.



WIRING THE WIRES AND WEAVING THE LOOM

The innovations of Babbage and Lovelace became the blueprint for modern computing. Their vision of a programmable machine inspired generations of scientists and inventors to transform theory into reality, leading from mechanical looms and engines to electronic computers. What began as an effort to automate patterns of thread became the foundation of automating patterns of thought a weaving of logic into machinery that ultimately gave rise to the digital age.



DEBUGGING THE BUG

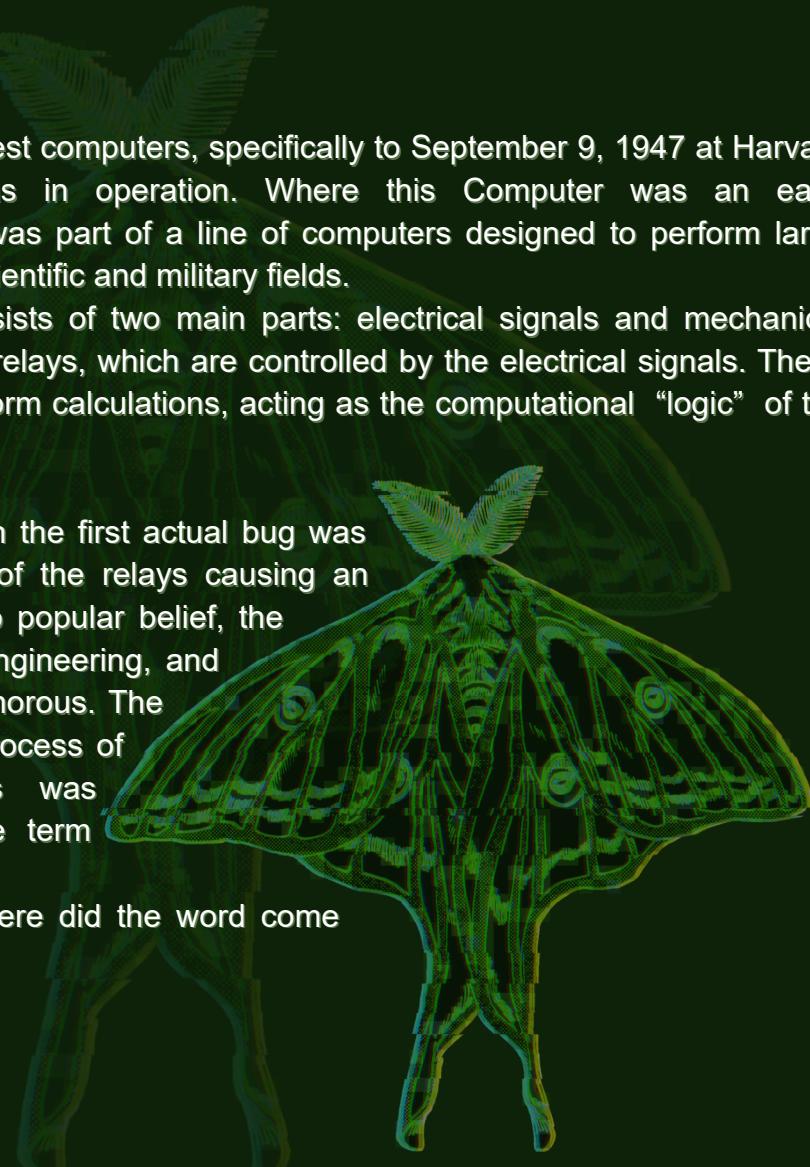
In the world of computing and coding, we may be diverse in paths, but we are united in the errors we face in codes, or as we refer to it bugs. But why do we call it bug in the first place, why not something else?

Let us travel back to one of the earliest computers, specifically to September 9, 1947 at Harvard where the Mark II computer was in operation. Where this Computer was an early electromechanical machine, which was part of a line of computers designed to perform large scale calculations automatically in scientific and military fields.

An electromechanical machine consists of two main parts: electrical signals and mechanical moving parts such as switches and relays, which are controlled by the electrical signals. These relays would open and close to perform calculations, acting as the computational “logic” of the machine.

Going back to Mark II , it was when the first actual bug was recorded, a moth, trapped in one of the relays causing an unexpected malfunction. Contrary to popular belief, the term “bug” was already in use in engineering, and the Mark II incident was largely humorous. The word “debugging” referring to the process of identifying and correcting errors was popularized by this event, but the term “bug” itself was not.

So we face the question again, where did the word come from?



The Thief of Inventions

Lets go even further back, in the late 19th century, when Thomas Edison frequently used the term “bug” in his notebooks to describe flaws or glitches in his electrical and mechanical inventions. In an 1878 letter, for example, he referred to “bugs” in electrical circuits that needed to be fixed. This is one of the earliest documented uses of the term in a technological context, showing that engineers were already using “bug” to describe troublesome faults long before computers existed.

The Horror

Interestingly, long before being used by engineers, by the 17th–18th centuries, “bug” was used in English to describe small insects, particularly ones that were pests or annoying, a label used broadly for small pests, including moths, flies, and beetles.

Going back even earlier, in Middle English around the 14th century, the word “bug” actually has roots in the word “bugge”, meaning specter, goblin, or frightening object, related words such as bogeyman, bugbear, bugaboo, and boggart, all stem from this root. In this early context, a bug was understood as an unseen disruptor, capable of causing fear, misfortune, or mischief.

The Bugging Bug

It is quite spectacular how the word evolved from meaning a mischievous spirit, became an insect and then later on found its way as a synonym of an error. Yet, one thing that didn't change is the fear and disruption that the word “bug” carried through the centuries whether in folklore, nature, or code.



THE MACHINE CONSUMES TO SPIT OUT MUSH

WHILE PATTERNS ARE CHASED AND BARELY UNDERSTOOD

IDEAS SPIN OUTWARD, LOOPING, FOLDING, AND COLLIDING, UNTIL
IT SPINS INWARD, ITERATING, DOUBLING, AND DIFFERING

SOFT THREADS, LIKE SILK, DELICATE FABRIC AS LACE,
LIKE A SPIDERS WEB, A NETWORK TO BE SET

WHAT DOES ONE CREATE, WHEN ONE FOLLOWS CURIOSITY INSTEAD OF CONVENTION?

A QUESTION TO BE WHISPERED BY ALL

BUT THE MOTION DOES NOT STOP. THE VORTEX PULLS ONWARD, AND THE BOUNDARIES
PERCEIVED BETWEEN COLLIDING MEANINGS, AND THE CONTRADICTION OF ITS OWN,
CHANGING LIKE THE OCEANS WAVES AND TIDES.

PERHAPS WE ARE NEVER MEANT TO SEE THE CONCLUSION, THE REWARD, THE END...
BUT TO LIVE A JOURNEY, SPINNING WITHIN AND OUTSIDE THE CIRCLES OF OUR OWN

PERCEPTION

