

>wired\_

**THE  
ZINE**

Edition 1

# TABLE OF CONTENTS

Interview with Former EY Tester.....	1
Why Coding Assignments Aren't as Hard as you Think.....	6
How Mixed Reality is Set to Change Our Lives.....	17
Introduction to Gaming Laptops and PCs for Dummies.....	22

# CREDITS

Interview with Former EY Tester:  
Leo Whitehead

## WRITERS

**WHy Coding Assignments Aren't as Hard as you Think**  
*by Surya Sathyamurthy*

**How Mixed Reality is Set to Change our Lives**  
*by Ngoc Han Lu*

**Introduction to Gaming Laptops and PCs for Dummies**  
*By Gina Lam*

## DESIGNERS

*Jasmine Subroto*

*Rahul Pejathaya*

# INTERVIEW WITH FORMER EY TESTER

## Interested in Cybersecurity?

*Curious about what it is like to be a Penetration Tester at EY?*

*Join us as Leo Whitehead, a Computer Science graduate from Monash, walks us through his journey of breaking through the niche field of Cybersecurity!*

**- Please introduce yourself and tell us a bit about your background!**

My name is Leo and I'm an ex-Monash student. I graduated from the Bachelor of Computer Science in 2023. Up until the end of last year, I worked as a penetration tester at EY within their Advanced Security Centre.

During my studies, I was mainly involved with MONSEC, Monash University Cyber Security Club. At the end of my first year, I joined as a Member Training Officer, as a Gen Rep. The year after that, I became the vice president of the club. As a member training officer, my main responsibility was to organise weekly workshops, find guest speakers and people from Monash

Monash that were willing to talk about various topics. However, as is the case with most people in that role, I wasn't able to find enough guest speakers to fill every workshop slot, so I ended up presenting most of the workshops myself. It was a lot of fun. I certainly learned a lot from it, but I ended up feeling quite burnt out by the end of it.

I think taking on that responsibility is what taught me the foundation of most of what I learned about cybersecurity. It definitely made a big difference in terms of my career afterwards. So I'd say it was a great opportunity to do that originally.

**- Do you think that being involved in these extracurricular and club activities had a great impact on you and your career path?**

It definitely did. Being able to present workshops on a regular basis and talk to large groups of people is something that I feel like doesn't come naturally to me. It takes a bit of practice, so running these workshops on a regular basis taught me to [develop] better skills of presenting in front of crowds and better skills in terms of preparing presentations.

And that's something that I did occasionally at work [as well]. I occasionally gave internal presentations or I took charge in some meetings.

So I'd say that it definitely made a big difference in terms of actual work skills. I would recommend everyone to get involved in as many extracurriculars and club activities as they can.

To some extent, I gained stakeholder skills through the roles at MONSEC and [provided] me a good amount of experience. Stakeholder management is [actually] a very big part of the job as a penetration tester. You know, keeping clients up to date with what the status of a test is, what vulnerabilities have been found, and communicating results to them. So good stakeholder management makes a big difference and is a very important skill.

**How did you [actually] find MONSEC in the first place? Did you find them a stall? What was your process?**

Yeah, I think I first heard about Monsec at one of the Open Days a few years before I went to Monash. I met a member of the committee (who is later the president) at the time. But afterwards, when I [started studying at] Monash, I had a bit of an interest in cybersecurity. I didn't know what I wanted to specialise in yet. I had a passing interest in a lot of different topics, and so I kind of sought out anything cybersecurity related.

When I kind of recalled that there was a cybersecurity club and I saw it on the MSA website, I kind of went and made sure to check out the workshops and just see what the club had going on.

I'd say the two big things that influenced my career the most were definitely getting involved with MONSEC, as well as the industry-based learning program, which I did my second last year. Penetration testing is a very technical role compared to many other roles, even in Cybersecurity. So, [participating in] those extracurricular activities, outside of the Monash units

the biggest difference overall.

- Would you say that the practicality of what you learnt at MONSEC, or during IBL, that really pushed you to the profession?

It is the practicality. I think just most of what's taught at Monash is not quite deep enough because ultimately the industry moves quite quickly and it's difficult for a unit to stay up to date. On top of that, cybersecurity is a bit of a cat-and-mouse game between attackers and defenders. When a vulnerability is found, people publish techniques for preventing that vulnerability, then attackers find new workarounds for that and then you have to find ways to patch those workarounds.

There's a limit to how in-depth you can get at uni and so they only really teach the foundations, and the foundations are definitely important, but I think it is [also] important to do a lot of your own self-study and to get into the role. Certainly, all the good testers that I worked with were genuinely passionate about the subject and spent a lot of time researching it in their own time.

**How did you end up working at EY? Was it through IBL?**

Yeah, it was through IBL. So I did a semester-long placement at EY. Initially, my placement was at Cyber Security Consulting, but in a relatively non-technical field. I was in a general consulting team that did GRC (Governance, Risk and Compliance) work or governance and regulatory compliance mostly, as well as mostly just PowerPoints, Excel spreadsheets, very generic consulting.

However, my goal was to transfer into the Advanced Security Centre, which is where penetration testing and some more well-thought-to-be-interesting roles were available

the last few weeks of the placements after doing some kind of technical tests within the placement so they could kind of see what my background was and see whether it would be a good fit for the team. After that, I joined the team and they offered me a part-time role, and so I continued working there for about a year while I continued working there casually for about a year while I completed my studies, and then after I finished my studies, I then joined full-time and worked there up until November last year.

**- You were talking about the transition between general consulting into the Penetration Testing role, and you mentioned that a major part of this was showcasing that passion and being on top of the 'cat and mouse' race. How were you able to transition that?**

Having a passion is probably the most important thing because with this being a cat and mouse game, one of the responsibilities in penetration testing is that you need to be constantly learning new things. You need to be constantly doing certifications, undergoing new training, learning about new systems, expanding your knowledge just because the threats that are out there are constantly changing.

So from an employer's perspective, the main thing they keep in mind [and consider] when they're looking at new people to hire is that you're capable of learning new techniques. They know that when you join, you're going to take a while to kind of learn about the job. You're an investment really, so they need to know that [...] you can continually go through training and keep on top of the industry trends as they change.

So demonstrating a genuine passion for the subject is probably the biggest thing you can do in terms of advancing your career. Being able to demonstrate

e that you, say, participate in CTFs or that you do hack-the-box machines or that you contribute to open-source software relating to cybersecurity or really anything relating to just showing that it's something that you're really interested in is probably the main thing that employers are looking for.

They understand that most people directly out of university probably don't know the most about the subject. They definitely don't know as much as a senior consultant and they probably don't have the industry experience yet, but taking you on, kind of investing in your future, in order for them to do that, they just need to know that you're someone that will put in the work on your side as well.

**- Have you ever made any mistakes that you think might have propelled you to grow or mature?**

Yeah, I would say the main mistakes I probably made when I was initially [in] my role was not asking enough questions. Since these jobs are often so competitive to get into, and because everyone seems so driven and passionate in the roles, I think admitting [and] saying that you don't know something and asking questions is something that a lot of people don't do enough of.

I know that when I started, I spent a lot of time trying to figure things out myself, reading internal wikis and Googling everything instead of just asking stuff to the people next to me. I [eventually learnt] that when you're starting a new job, especially in the first month, people expect you to ask a lot of questions and people around you are very willing to answer most of them.

I have [also] seen other people fall into that trap. I'd say that the main thing to keep in mind is that cybersecurity is a team sport and ultimately a lot of the

most interesting vulnerabilities I've found on the job have come as a result of just bouncy ideas of other people and just having a chat and asking them for their opinions on where vulnerability might lie or what piece of functionality I should look at. The main thing to just keep in mind is that it's all very collaborative effort and everyone's better off in the long run, everyone just kind of keeps that in mind.

**- What is your favourite thing about being a Penetration Tester or just working in the field in general?**

Being able to say that you get paid to hack people is a lot of fun. It's certainly a good conversation starter. I think ultimately my favourite thing about the industry or about this role specifically is that it rewards people that have very deep knowledge and kind of deep technical understandings of many different systems. At its core, a vulnerability arises when an attacker understands the capacities of a system better than the engineer that developed it – So it's all about understanding various different unrelated technologies at a very intuitive level.

I mentioned earlier that I was a bit interested in cybersecurity when I started but I was interested in a lot of other different topics. I didn't really know what to specialise in, I enjoyed low level programming, I enjoyed cryptography, I enjoyed compilers and language development and I just didn't really know what to specialise in.

But ultimately I think what pulled me into cybersecurity is the fact that it rewards people that kind of have all these different interests. Really there's no topic that if you want to spend a week doing a deep dive into some random topic, there's really no topic that you could learn that would be considered a waste of time. There's always some applicability to cybersecurity

if you can understand something else at a deep level.

**- Do you have any advice for students interested in your field?**

Definitely [having soft skills like] communication ... and to an extent stakeholder management as well, because half of your job is performing penetration tests and finding vulnerabilities, but the other half is [actually] writing reports and communicating those vulnerabilities to other people. If you can understand some incredibly niche technical high level vulnerability that causes a massive amount of damage to systems, that's cool. But if you can't communicate that clearly to an executive team without a technical understanding of the business implications of that vulnerability, then it's practically pointless even knowing about it. Being able to communicate well, and present yourself well, is probably the most important bit of the job, [maybe] even more important than the technical skills to be honest.

Also, networking. Cybersecurity is a relatively niche field, especially amongst Penetration Testing, and a lot of people know each other, so networking probably does help a great deal. I tried to network to some degree when I was early in my career. I probably haven't networked as much as someone that's been in this job for longer has, considering I've only got about a year and a half of experience doing this, but someone working for longer than that would probably have a larger network at this point.

If you're interested in networking in terms of cybersecurity in Melbourne specifically, I'd recommend going to the regular monthly meetups such as Ruxmon and SecTalks Melbourne, and maybe attend some very cybersecurity conventions because there's a lot of interesting people in the city, so getting to know a handful

is very useful.

Also when I was transferring teams between the non-technical role into the Penetration Tester role, the fact that I knew a handful of people in that team that could vouch for me made a very big difference in terms of my ability to transition between the different roles.

In fact, at the time, my involvement at the club was mainly what got me those networking opportunities because a few of the testers working in the Advanced Security Centre at the time were actually on the MONSEC committee prior to me. Having that shared connection there definitely made a big difference.

**- How do you think you go about making a good impression for people, especially if they want to take you on as a protege or vouch for you? How do you go about establishing that for yourself?**

Just showing an interest in the subject. As I was discussing before, universities generally don't teach you penetration testing terribly well [because it is quite a specialised role]. So most people that go into this kind of role do so because it's something that they spend a lot of their free time researching into and reading up on.

It's usually a topic that they're passionate about. So if you can demonstrate that to someone, you [can] immediately form a strong connection [and] share such a key interest with someone. So having something in common helps a lot.

[I noticed that] showcasing the passion plays a big role in being able to establish that connection with the other person. Just generally just having common interests as people. Discussing the area of things. in the industry, discussing the kind of recent news and happenings and things.

**- You mentioned showing passion is quite important. Do you have any recommendations for students to spark that kind of passion?**

Well, for people [who are interested in Cybersecurity] but have not joined MONSEC [yet], join MONSEC [and clubs of interest]. I may be biased there, but that definitely was helpful for me. But also, if you're interested in learning more about cybersecurity, I'd recommend participating in CTFs or capture the flag competitions.

I mean, everyone gets into cybersecurity in completely different ways. Most of the people I worked with all had very different stories in terms of how they got into the field. Some of them worked in system administration before they came over, some of them used to be software engineers, some of them used to be plumbers. So everyone kind of came into it differently.

My background was through CTFs. They're competitions in which you earn points for hacking intentionally vulnerable websites and you get more points depending on how difficult it is to hack. For me, the main push into the field for me was joining MONSEC's competitive CTF team, and I had a lot of fun there.

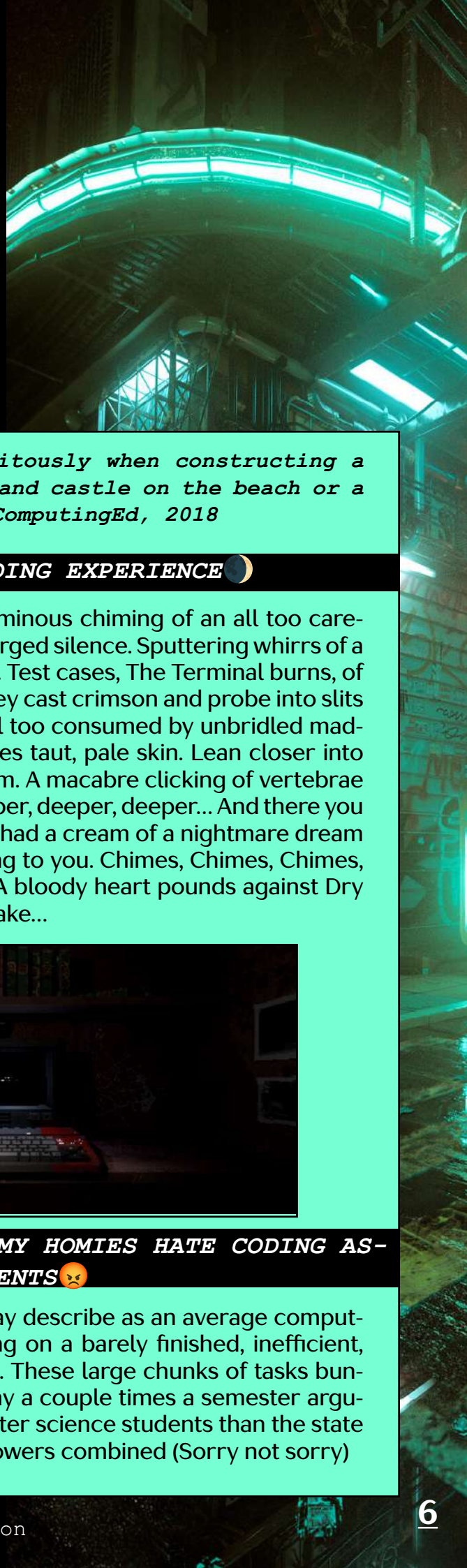
But ultimately, if it's something that you're interested in getting into the field, or if you're not sure what field you're getting into, just continue learning whatever topics you find interesting because that will be relevant to whatever you want to do in the future, whether that's cybersecurity or something else.

***Special thanks to Leo Whitehead, Rohan and RADMON for facilitating this interview.***



# WHY CODING ASSIGNMENTS AREN'T AS BAD AS YOU THINK

By Surya Sathyamurthy



*"Learning occurs most felicitously when constructing a public artefact, whether a sand castle on the beach or a theory of the universe." - ComputingEd, 2018*

## 🌙 A MIDNIGHT CODING EXPERIENCE 🌙

The sleeper has awakened. The ominous chiming of an all too careless bedroom clock pierces submerged silence. Sputtering whirrs of a computer fan cough oil and grime. Test cases, The Terminal burns, of those blood-stained-scriptures, they cast crimson and probe into slits that once held pupils of a being all too consumed by unbridled madness. A foul wretched, reek escapes taut, pale skin. Lean closer into the horrors, bare witness to a doom. A macabre clicking of vertebrae ensues as your spine contorts deeper, deeper, deeper... And there you are, you ghoul of the night, you've had a cream of a nightmare dream and you've got the hoo-ha's coming to you. Chimes, Chimes, Chimes, a Harbinger of Inexorable Time... A bloody heart pounds against Dry Bones... Fear the Maker and His Wake...



😡 **F\*\*\* ASSIGNMENTS, ALL MY HOMIES HATE CODING ASSIGNMENTS** 😡

The passage above is what you may describe as an average computer science student's night, working on a barely finished, inefficient, non-functional coding assignment. These large chunks of tasks bundled together and thrown your way a couple times a semester arguably cause more anxiety to computer science students than the state of the job market, inflation and showers combined (Sorry not sorry)

Some coding tasks can be so laughably obtuse and difficult with pedantic marking schemes that even the most resilient among us struggle to find the willpower to complete them. One would not be wrong to question, why are universities across the world still assigning these projects to “teach” content? Surely there must be a better way for us to learn these important and, frankly, very engaging and interesting concepts, so that we can succeed in job interviews and stoke our passion for programming.

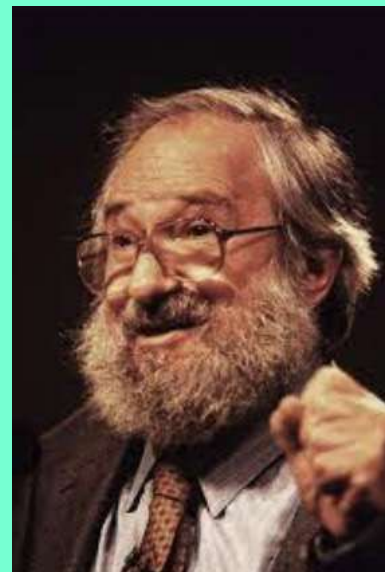
I too believed in the sentiment that coding assignments were torture devices that emerged from the mouths of Shai-Hulud (Can you tell I like Dune 😊). However after some in-depth research on pedagogy, and learning philosophies birthed from the bathtubs of ancient greece philosophers and the dusty M.I.T computer labs of the 60’s, I have begun to appreciate the structure of coding assignments and the manner in which they teach us important concepts.



*Captured here is Marvin Branagh of Racon Police Department seeking Leon's help to debug his failing code (he was brought to the brink of death by the difficulty of the coding task he was assigned.*

## 🕹️ INTRODUCTION TO CONSTRUCTIONISM (WITH AN "S") 🕹️

The reason I am playing this devil’s advocate role () is because I believe coding assignments at Monash University follow a “Constructionist” method of teaching. The theory of constructionism describes “student-centred, discovery learning where students use what they already know to acquire more knowledge.” (Wikipedia, 2024). This learning theory was established and heavily worked on by Seymour Papert; computer scientist, mathematician, artificial intelligence pioneer, MIT professor (Britanica, 2024) and the man who developed constructionist learning, based on the theory of constructivism through the teachings of pedagogist (a word describing an expert in the theory and methods behind learning acquisition, especially in early adolescence in our Seymour’s case) Jean Piaget.



*Seymour Papert 1928-2016*

Constructivism (spelt with a “v”, unlike constructionism by Papert. I know they’re quite similar so it’s confusing, but I didn’t come up with the names so sue me I guess 🤔), is the idea that “humans create knowledge through the interaction between their experiences and ideas” (EdTechBooks, 2024) and that throughout all stages of learning acquisition, “the individual is at the centre of the knowledge creation and acquisition process” (EdTechBooks, 2024). I won’t delve too deeply into constructivism as it’s quite a deep rabbit hole, all that is important to know is Papert’s interpretation and expansion of Piaget’s ideas through computer science and programming.



*Seymour Papert in an instructional video teaching recursion through Logo*

Combining Piaget’s constructivist principles with the rapid advancement of the “learner’s own artefacts and technology” (referring to programming and one’s ability to use it to learn effectively), constructionism “promot[es] learning experiences and empower[s] students to create and construct knowledge through [their] projects’ interaction with the world” (Edith Ackermann, 2001)

My goal is to highlight that humanity’s ways of learning, communicating knowledge through conceptual models, experimental thinking, and group collaboration can all be translated into programming. Our tools and artefacts (as Papert likes to call them) have evolved so much that we have created a technological language to express ideas. Therefore, I claim coding assignments, when approached meaningfully and with a desire to actually perform and learn, is an incredible way to learn programming/mathematical concepts.



## 🤖 HOW CODING FOLLOWS PAPERT'S INSTRUCTIONISM 🤖

I can objectively prove (🤖) our assignments strictly follow Papert's philosophy for engaging:

1. Tasks should be related to a larger idea/project
2. The learner needs to be supported
3. Allowing learners to experiment with their results "against different views in different contexts". (Wikipedia, 2024)

Are these not all aspects represented by the coding assignments we claim to be the bane of our existences? Addressing each of these rules in the context of assignments:

### 1 *The assignments are related to an interesting larger project*

(eg. Connect 4, Simulating Pokemon Battles, Creating a Database for an Art Gallery System) consisting of many separate tasks. Despite their difficulty, Papert still believed in a focus of "projects not problems, [and that students can] learn more if they learn concepts in the context of projects" (Mitchel Resnick, 2018). Papert also addresses the difficulty of coding projects, stating that: "through [the] construction of tasks towards a [large idea/project], students will face complex issues, and they will make the effort to problem-solve and learn because they are motivated by the construction". (ComputingEd, 2018).

### 2 *Lecturers and Youtube channels teach content while the bulk of the assignment is given to you to solve*

with TA's taking a mentoring role. A core principle of Papert's teaching is engagement, not explanation, hence he advocated for a more flexible role of teachers; they don't teach directly towards students, but instead support them.



*Leetcode my beloved...*

**3** *We run our code against staff and test cases to prove its adaptability, correctness and functionality.*

We may also use this code we've written for outside university purposes or maybe to inspire our own version of the assignment. This relates to Papert's belief that "that students will be more deeply involved in their learning if they are constructing something that [themselves] and others will see, critique, and perhaps use" (ComputingEd, 2018).

The feeling of being thrown into the deep end by coding assignments, while seemingly unfair and needlessly evil (😈), objectively satisfies all prerequisites to learn effectively in terms of constructionism and Seymour Papert. Does this mean students in the IT faculty are better than every other student at Monash simply because programming is an amazing manner in which to facilitate learning and understanding 🤔?



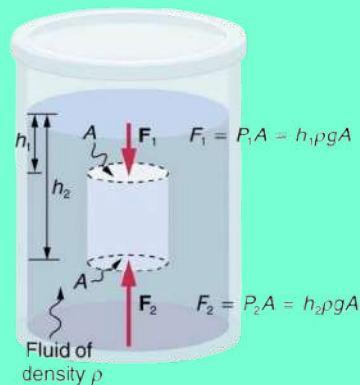
## PAPERT'S CONSTRUCTIONISM IN ANCIENT GREECE AND THE ORIGIN OF ARCHIMEDES' PRINCIPLE

Reflecting on learning methods through history, the principles of constructionism can be traced back to, for example, Greek philosophers such as Archimedes and his method of measuring volume through the displacement of water (indulge me for a second, I promise this is going somewhere 🙄🙏). The most simple, logical and universal consistency in the world is that water overflows out of a tub if you immerse yourself in it while it's full. Despite its apparentness however, Archimedes indulged in his innate sense of curiosity and was able to piece together that the amount of water displaced depended on how much of his body was immersed. He was so ecstatic about his discovery that he leaped out of his tub and ran naked in the streets of Syracuse yelling "Eureka!" (TedEd, 2023).



Graphic from TED video on Archimedes Principle (the link for full YouTube video can be found in the full bibliography)

A formula written on paper is not digestible by itself Papert argues; "all of us learn by constructing, exploring, or theory building". (Amy J. Ko. 2017). Simply observe the diagram to the right for the displacement of water:



It's gibberish and messy right?! But this complicated diagram describes the universal certainty of Archimedes sitting in his bath and water overflowing. Papert argues that only when we are able to contextualise concepts in our own, personally manufactured manner, with our watermark splattered all over our work (no matter how jumbled together, objectively inefficient or easily criticised) can we truly understand what the concepts we are being taught mean in the physical world. Only then should we begin to formalise our interpretations.



*Archimedes approaching the King telling him he overflowed the bathtub and can now prove if the blacksmith cheated him (circa 246 colorised) - watch the Ted video to understand*

short 3 minute video 🤔).

All IT students have had their Archimedes Eureka! moment: creating their very first hashmap to solve a LeetCode problem, implementing working left and right pointers, visualising the upper and lower bounds of an algorithm. These well known concepts were all conceived in a personalised and conceptual manner. Only after a personal understanding of these concepts through the mentoring of a TA/friend/YouTube could you link your prior/current knowledge to associated formal representations (such as Archimedes formalising overflowing bathtub water through an equation measuring the displacement of water).



*How IT students feel after finally sinking hours to solve a difficult leetcode problem (it was TwoSum)*

We have a huge leg up on Archimedes though: Programming! After these Eureka moments, we can piece together our understanding in the material world and throw as many test cases we want at it! We literally put together bricks of understanding in the language of code, we can express our personalised interpretation of a mathematics/programming concept. Our euphoric moment is now captured in time forever in a testable, critiqueable, materialistic manner. We have theory-built, experimented and collaborated to create our own understanding of a concept and brought it to life.

Is this not what Papert has been discussing to be the perfect form of learning? We use our genuine curiosity to foster “discovery learning” and investigation, then we experiment/mould the concept through coding, and finally we use this code as part of a larger,



more interesting project. We have followed his steps for perfect learning without even realising it! We now have hundreds of “Logo” programming languages these days to choose to repeat this process and learn about all aspects of programming concepts, geometry, mathematics, etc...

*“The child programs the computer and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building”*

*-Seymour Papert, Mindstorms*

## 🤖 THE HUMAN INSTRUMENTALITY PROJECT 🤖

*But instead of all humans ascending to transcendental levels of existence where they exist as a singularity, it's my arguments I've made in this article transcending into singularity. So, essentially, a conclusion*

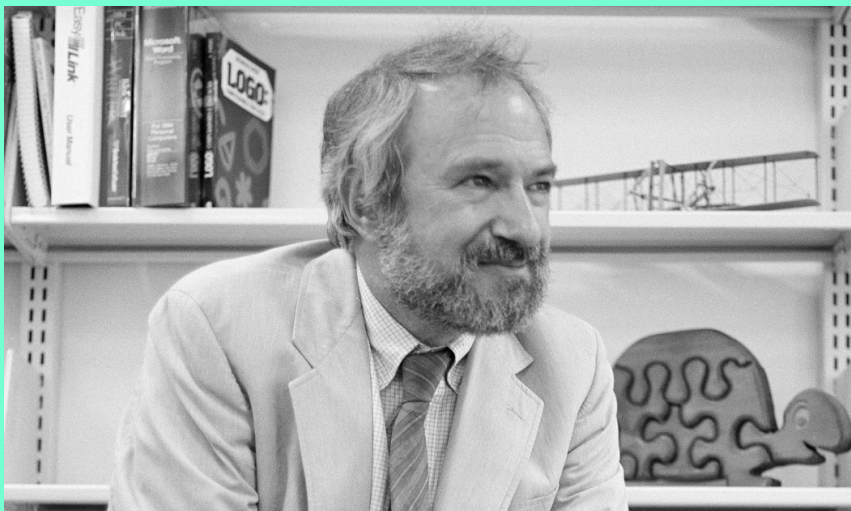


As Professor J. Ko of the University of Washington describes, Papert wants us to build knowledge “literally out of bricks, blocks, code, or paper [and] that our process of learning as humans is an iterative, cumulative one, not one of “absorbing” knowledge fully formed” (Medium, 2017). We have to take time to digest and interpret ideas before formalising them.



It can be argued that coding assignments are not the most flexible assignments, however isn't that all the more reason to start your own coding projects? Using Papert's philosophies discussed in this article of engagement, experimentation and a child-like approach to learning, you can create anything and have a great time doing it. These interesting coding/mathematics concepts are not being taught "poorly" in university, and this dreaded process of trudging through task after task is actually rooted in very solid learning theories from well renowned pedagogy experts.

Have faith that through being thrown in the deep end, being forced to essentially learn coding through coding, is how we learn and immerse ourselves in computer science; the science that our every waking moment relies on. Trust in the process and trust in the late Seymour Papert, a computer scientist just like us, whose passion and work for education solidified the role of computers in education forever.



*"Let a child learn mathematics by speaking in mathematics about things that really matter to him"*

*- Seymour Papert*

## REFERENCES

1. Wikipedia. (n.d.). Constructionism (learning theory). Retrieved from [https://en.wikipedia.org/wiki/Constructionism\\_\(learning\\_theory\)](https://en.wikipedia.org/wiki/Constructionism_(learning_theory))
2. Wikipedia. (n.d.). Constructivism (philosophy of education). Retrieved from [https://en.wikipedia.org/wiki/Constructivism\\_\(philosophy\\_of\\_education\)](https://en.wikipedia.org/wiki/Constructivism_(philosophy_of_education))
3. Mitchel Resnick. (2017, March 22). Mitchel Resnick - On Seymour Papert. Retrieved from <https://www.youtube.com/watch?v=ZoczAscGYeQ>
4. Seth Morabito. (2009, June 19). Seymour Papert. Retrieved from <https://www.youtube.com/watch?v=xMzozQFyMo0&t=190s>
5. Mitchel Resnick. (2022, 23 Feb). Engagement not Explanation. Retrieved from [https://www.youtube.com/watch?v=9qj1E-uhFYc&list=PLhSebWQzHLEuxRJR5Z\\_8Oc91fbOH3dS9O](https://www.youtube.com/watch?v=9qj1E-uhFYc&list=PLhSebWQzHLEuxRJR5Z_8Oc91fbOH3dS9O)
6. Amy J. Ko. 2017, July 13). Mindstorms: what did Papert argue and what does it mean for learning and education. Medium. Retrieved from <https://medium.com/bits-and-behavior/mindstorms-what-did-papert-argue-and-what-does-it-mean-for-learning-and-education-c8324b58a-ca4>
7. The Daily Papert. (2024). About Seymour Papert. Daily Papert. Retrieved from <https://dailypapert.com/about-seymour-papert/>
8. Bekki Brau. (2024). Title. Constructivism. Retrieved from <https://edtechbooks.org/studentguide/constructivism#:~:text=Jean%20Piaget%20is%20known%20as,between%20their%20experiences%20and%20ideas>
9. Computing WordPress. (2018, March 19). Constructivism vs. Constructionism vs. Constructivism. Computing Education Blog. Retrieved from <https://computinged.wordpress.com/2018/03/19/constructivism-vs-constructivism-vs-constructionism/>
10. Britannica. (2024, February 25). Seymour Papert. Retrieved from <https://www.britannica.com/biography/Seymour-Papert>

11. Edith Ackermann. (2001, January). Piaget's Constructivism, Papert's Constructionism: What's the difference? ResearchGate. Retrieved from [https://www.researchgate.net/publication/238495459\\_Piaget's\\_Constructivism\\_Papert's\\_Constructionism\\_What's\\_the\\_difference](https://www.researchgate.net/publication/238495459_Piaget's_Constructivism_Papert's_Constructionism_What's_the_difference)

12. Misha Ketchell. (2016, August 25). This little known pioneering educator put coding in the classroom. The Conversation. Retrieved from <https://theconversation.com/this-little-known-pioneering-educator-put-coding-in-the-classroom-63971>

13. Mitchel Resnick. (2023, Feb 13). Papert: Early Logo Explorations. Retrieved from <https://www.youtube.com/watch?v=-QgcP8KkFdc>

14. TED-ED. (2012, September 7). How taking a bath led to Archimedes Principle - Mark Salata. Retrieved from <https://www.youtube.com/watch?v=ijj58xD5fDI>



# HOW MIXED REALITY IS SET TO CHANGE OUR LIVES

By: Ngoc Han Luu



Source: How AR & VR are Revolutionising the Gaming Industry ([thegamemarketer.com](http://thegamemarketer.com))

In a world of advancing technology, the entertainment industry stands at the forefront, reshaping how we consume content and experience entertainment. As Augmented Reality (AR) and Virtual Reality (VR) continue to evolve into different applications of technology, this sparks curiosity for human-discovery on how the digital sphere can harmonise with our day-to-day life. Yet, as we embrace these innovations, we must acknowledge that there are social challenges and societal adjustments that arise from the unique problems introduced by new technology.

To understand these challenges further, we first need to understand what are the fundamentals of the technology behind AR/VR and how industries have planned or utilised these into their business and products.

## 🤔 What is AR/VR and Mixed Reality? 🤔

Augmented Reality (AR) and Virtual Reality (VR) are two distinct but related technologies that immerse users in digital environments. But how do they differ?

### **AUGMENTED REALITY**

AR uses computer technology to generate a simulated environment where users can completely immerse themselves within a 360° setting. Through overlaying digital information onto the physical world, AR enhances the user's perception of reality with additional virtual elements (i.e sounds, graphics



or data) to the user's surroundings (Guides Library Utorono, 2021). This functionality can be accessed through various devices such as smartphones, tablets, or smart glasses.

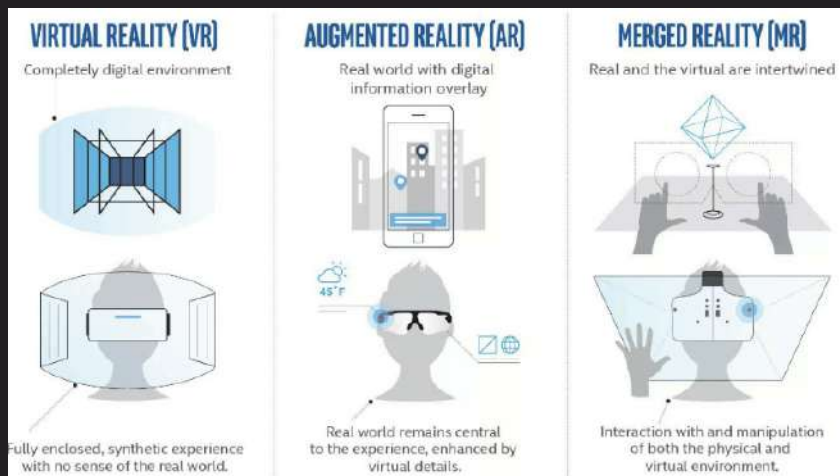
## VIRTUAL REALITY



Whereas for VR, your whole complete environment is immersed by what a computer has generated for your world. Comprising a VR headset that blocks out the physical world, it would typically replace it with a simulated world instead.

## MIXED REALITY

Then we have Mixed reality; which is really just a combination of both! However, the main difference is that mixed reality merges both the digital computer generated world with reality, allowing both worlds to coexist. Let's dive deeper and see where mixed reality really plays a significant role.



Source: *Augmented Reality vs. Virtual Reality vs. Mixed Reality* | Toptal®

Then we have Mixed reality; which is really just a combination of both! However, the main difference is that mixed reality merges both the digital computer generated world with reality, allowing both worlds to coexist. Let's dive deeper and see where mixed reality really plays a significant role.

## 🎮 ENHANCE YOUR GAMING EXPERIENCE 🎮

If you're familiar with Minecraft, (a story based game that allows players to explore and create a life in a digital world), then you'll know how thrilling the adventure is by just simply using simple controls to move around to explore your world. But what if you just were the actual character and just by moving your head, you'll get to discover the world around you? Sounds cool right?



With AR and VR technology, these features enhance the user experience so that it feels like your whole body is the character exploring a digital world. Other than the visual aspects, AR/VR games ensure that you utilise most of your senses such that a tracker will detect a user's movements and have assisting audio and haptics to come off as if you are "the character" in the game itself (Pwc, 2022). Additionally, combining these technologies into these games will also send visual and audio feedback, creating an ideal gameplay environment for players making it more exhilarating to play video games.

On a further note, the success in these stems from the hype of this immersive gaming experience. In 2022, AR games acquired a market share of \$8.4 billion (300 Mind Studio, 2023). There's a huge potential for the gaming industry like adventure/action games, stimulations, educational, sports and much more!



## 😬 SOCIAL BARRIER CHALLENGES 😬

Despite all this, there are some challenges that come along with the adoption of mixed reality especially when there isn't a distinct line that Mixed reality should be limited to. In a recent case of Apple's new headset product, Apple Vision Pro, it has sparked a sense of curiosity as well as hesitation in where the future could potentially lead.

A brief social experiment conducted by Arun Rupesh Maini (youtube: @Mrwhosetheboss) showcases a mixed reaction after observing how people would react if the Apple Vision pro headset was worn casually in public. Issues arise from the physical barrier of the headset itself covering a person's facial features, especially many claiming that covering one's eyes with a digital headset creates difficulty in communication and focus. Despite Apple's effort to slightly tint an opaque window for the eyes to overcome this obstacle, many find it hard not "to stare at you" or simply dodge interactions as clearly, the headset was a clear elephant in the room.

## 👉 A HEFTY PRICE.. is it worth it? 👈

Another challenge that is faced is the current price tag. Although there are a few alternatives to the Apple Vision pro (i.e Meta Quest 3, Pimax Crystal and Varjo XR-4), the price ranges from \$900 - \$3,500.

You would expect that for a luxury item, the cost would be justifiable – with many apps that basically could replace your phones. However, an underlying struggle as for now is that not all developers will be able to get their hands on these headsets, and hence, it can be a constraint for developers to enter the market. As a result, as a consumer, there may be certain apps that we'll just have to wait (i.e Apple Vision pro doesn't have major apps such as Netflix, Spotify and Youtube) (Barr, 2024)



However, despite that, it is suggested that these new technologies are only the beginning. As Apple and many other companies are pushing these exciting technologies into the market, this sets the stage for developers to learn more about the environment and discover endless opportunities. While these headsets are quite pricey, hopefully in the near future there would be a simplified and affordable mixed reality device for general users to have a new reality beyond this world.

## 🔥 ARE WE READY FOR THE NEW DIGITAL FUTURE?? 🔥

Mixed reality, augmented reality and virtual reality are all quite exciting and new technology. We know that as technology evolves, we need to think whether we should accept this new societal norm as it gains popularity. Is it something that we know that it's bound to happen hence we should just embrace it? Or should we be concerned about this technological advancement? But setting aside the uncertainty, there's a larger world that's waiting to be explored!

## REFERENCES

- Barr, K. (2024, January 22). Here Are All the Apps Coming to (And Not Coming to) the Vision Pro. Gizmodo.  
<https://gizmodo.com.au/2024/01/here-are-all-the-apps-coming-to-and-not-coming-to-the-vision-pro/#:~:text=A%20few%20big%2Dname%20apps,Vision%2Dspecific%20version%20on%20release.>
- Gibbs, A. (2022, October 5). Sensing the metaverse with cutting-edge tech. Pwc. Sensing  
<https://www.pwc.com.au/digitalpulse/sensing-metaverse-tech.html> the metaverse with cutting-edge tech (pwc.com.au)
- Maini, A. "Apple Vision Pro Review - Why does this EXIST?" Youtube. Uploaded by Mrwhosetheboss, 2 March 2024.  
Apple Vision Pro Review - Why does this EXIST? - YouTube
- Rajput, A. (2023, July 25). How AR & VR in Gaming is Transforming Gameplay Experience? 300 Mind Studio.  
<https://300mind.studio/blog/ar-vr-in-gaming-industry/>
- Tall, T. (n.d). Augmented Reality vs. Virtual Reality vs. Mixed Reality: An Introduction Guide. Toptal.  
<https://www.toptal.com/designers/ui/augmented-reality-vs-virtual-reality-vs-mixed-reality#:~:text=But%20mixed%20reality%20devices%20are,what%20we%20would%20normally%20see.>
- The Game Marketer (2022, February 25). Into the Metaverse: How AR & VR are Revolutionising Gaming. The Game Marketer.  
<https://thegamemarketer.com/insight-posts/into-the-metaverse-how-ar-and-vr-are-revolutionising-gaming>
- Wong, D. (2021, September 1). Research guides: Virtual Reality in the Classroom: What is VR? Guides library Utoronto.  
<https://guides.library.utoronto.ca/c.php?g=607624&p=4938314>
- Augmented Reality In Business: How AR May Change The Way We Work (forbes.com)



# 🎮 INTRODUCTION TO GAMING PCS & LAPTOPS FOR DUMMIES 🎮

By: Gina Lam

In the rare chance that you are like me, who is a complete computer noob despite studying IT 🤦, shopping for gaming PCs or laptops can be quite overwhelming when confronted with terms like “Intel Core XXXX processor”, “SSDs” and “GPUs”. It often leaves me feeling lost and floundering in the sea of technical jargon.



If you are feeling just like this, worry not for I am here to help you understand all the basics you need to know about computer hardware and components!

## 🎮 LAPTOPS VS PCS 🎮

Firstly, let's delve into the classic dilemma that many have grappled with throughout the years: gaming laptop 🖥️ or desktop PC 🖥️?

In most cases, many will recommend a gaming desktop. This is because not only are desktops easier (and less expensive 😞) to customise and upgrade, but they also run quieter and smoother than laptops — due to the small and slim fans which need to spin faster to cool down, hence noisier. Whilst a desktop use more power, and can add to your monthly power bill 💡, Smith (2023) notes that they generally outperform laptops in demanding situations and last longer.

Considering all this, I myself still bought a \$2.5K gaming laptop! A setback for my bank account, but for me, worth every cent!



While a desktop gives more bang for your buck, laptops provided me with something that a desktop cannot: portability. Being able to move my laptop around with ease was a dealbreaker for me when deciding between the two. What everyone may want or need with their gaming device will obviously vary, but once you decide, we can get to the technical part.

## ✨THE CPU ✨

The CPU, also known as Central Processing Unit, is basically the ‘brain’ of the computer. It is the core component that controls everything 🧠 including different applications on your computer like your browsers, Microsoft suite, etc., and functions like input and output. It communicates instructions to various units in your computer (eg. even something as simple as opening your Minecraft), and provides processing power that your computer needs to function properly.

There are two most well-reputed brands of CPU that you should consider: AMD and Intel. When deciding between these two, it is important to think about your 💰 brokenness budget 💰 and what you will use your gaming PC or laptop for because each brand has its own strengths and weaknesses.

For example, having a greater number of cores and threads allows your CPU to interact and handle multiple applications simultaneously, while having a higher clock speed contributes to an improved gaming performance (Shih, 2020).

Usually, AMD puts its focus on pushing for a higher number of cores and threads (Shih, 2020) while Intel provides for better gaming performance. However, these two brands have continuously produced improved and specialised CPUs over the years, and it is difficult to say which processor fits you best. For now, here is a little table comparing AMD and Intel in 2023 to help you get an idea of what may suit

vs Intel CPUs in 2023		
	Intel	AMD
ricing and	X	
g mance	X	X
nt on/Produc	X	
ications	X	
ocking	X	
mption		X
s and ire	X	
is Node		X
ecture	X	X
ty		X
r: Intel -	7	5

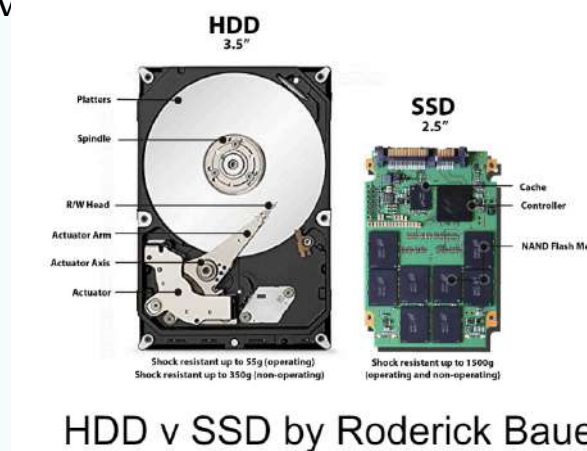
AMD vs Intel CPUs in 2023 by Paul Alcord

## ✨THE STORAGE ✨

Storage seems explanatory enough: it is where you 'store' and record all your data and software; this can include things that are not accessed regularly like photographs, word documents, saved game files, etc. It is essentially the 'garage' of your computer, and your saved data is your Lamborghini. There two storage drives that you should know about:

- (1) Hard Disk Drive ('HDDs'); and
- (2) Solid State Drive ('SSDs').

HDDs involve a spinning disc, which stores the data, and a mechanical 'read arm' that scans the disc to extract the information stored on it (CCL Tech, 2021). SSDs on the other hand have no moving parts and instead manipulates the states of



Given that SSDs' performance is better than HDDs, the former is expectedly more expensive than the latter. BUT, you must take note that while HDDs are slow, they are more affordable and can store much more data for the price (Hesse, 2022)

If you want to prioritise affordability over performance or vice versa, whatever you decide is all good as long as you don't suck at games and throw in the bottom lane when there is clearly a fight going on in the top lane (sorry, bad memories).

## ✨MEMORY (RAM) ✨

Now, I hear you asking, what is the difference between Memory and Storage?! oh wait, you didn't ask that? Oh. This is awkward. I'll just mansplain it anyway.

Put simply, storage saves files permanently while memory is temporary. RAM, or 'Random Access Memory', is used for short-term storage and only holds onto the necessary data as your programs run (Shih, 2020). Hence, if you want to run more programs concurrently on your computer, bigger RAMs are recommended.

Generally, 16GB and 32GB RAM is more than capable of doing the job. If you are a student like me and do nothing but play games, do work and complete projects on the side, 32GB+ RAM can be an overkill.

## ✨ GPU (GRAPHICS CARD) ✨

There is so much to talk about and delve into when it comes to this component. It indeed draws “the most attention from gamers” (CCL Tech, 2021) and might be the most expensive piece in your gaming device.

The GPU (or Graphics Processing Unit) is the “main silicon chip which sits at the heart of the graphics card” (CCL Tech, 2021) It renders in-game graphics and helps produce the stunning visuals that many games have.

The two major manufacturers for this component are AMD and Nvidia, and the rivalry between them is as fierce as ever in 2024. Webb (2023) revealed that typically, Premium Nvidia graphics cards are viewed “as the most powerful”, while the best AMD cards “have a reputation for being significantly more affordable and energy efficient”.

It is impossible to definitively say what the best graphics card is right now, ultimately, it is up to you, what you value and want for your gaming experience.

## ✨ MOTHERBOARD (MOBO) & CASE & PSU ✨

The motherboard is the main circuit board all the components plug into. It performs many important functions, including interfacing with CPU and other hardware. The case protects all of the hardware and holds all the components together. These components, for PCs in particular, are quite important to note as, for example, picking the wrong motherboard can “severely limit the features and upgrades” you might want to make in the future (CCL Tech, 2021).

I will not go too much into detail about these components but if you are interested in reading more, please refer to this website:

<https://www.pcguides.com/motherboard/faq/sizes-form-factor-explained/>

The Power Supply Unit (‘PSU’) is another central part of your computer’s power system. It “draws electricity from your outlet and powers each component” accordingly (Hesse, 2022). When building your own PC, the PSU is a critical component to consider because getting a PSU will not only protect your components, but also help you save on your energy bills through efficiency.

## ✦ CONCLUSION ✦

You can basically make your own computer now!! (Not really) But this is generally all you need to know about your computer because that's basically all of what your computer is made of (plus some wires for power, of course). Hope this helps you in your journey in buying gaming devices.

## ✦ REFERENCES ✦

- CCL Tech (2021, August 27). The Complete Gaming PC Glossary: A Guide to Terminology. CCL Tech.  
<https://www.cclonline.com/article/1891/Guide/CCL-Gaming-PCs/The-Complete-Gaming-PCGlossary-A-Guide-to-Terminology/#pc-hardware-and-components>
- Hesse, B. (2022, November 3). A Beginner's Guide to Buying Your First Gaming PC. LifeHacker.  
<https://lifelifehacker.com/a-beginners-guide-to-buying-your-firstbv-gaming-pc-1849735385>
- Kirkcaldy, A. (2023, January 30). Motherboard Sizes Explained. PC Guide.  
<https://www.pcguides.com/motherboard/faq/sizes-form-factor-explained/>
- Shih, H. (2020, September 24). A Noob's Guide to Gaming PCs — The Basics. Medium.  
<https://medium.com/@hunshih/a-noobs-guide-to-gaming-pcs-the-basics-2b1a161e4f40>
- Webb, K. (2023, August 31). AMD vs. Nvidia: How to choose the right graphics card for your PC. Business Insider.  
<https://www.businessinsider.com/guides/tech/amd-vs-nvidia-graphics-cards>