In elementary school I remember our teacher giving us challenge puzzles to solve on Friday afternoons when the day’s work was done. One day our teacher presented us with a puzzle so hard that it hurt my little brain just thinking about it. This puzzle was the Towers of Hanoi, and it was my favorite because I couldn’t find an obvious answer. To explain it briefly,

- a pyramid shaped stack of disks each one smaller than the other as it ascends needs to be moved according to a specific set of rules.

- A pyramid shaped stack of smaller and smaller disks is placed [placed together] on a peg. The objective is to move the stack to another peg following a specific set of rules.

We revisited the puzzle the next week, but sadly the confidence I had built up was quickly proven to be worthless. My once-hopeful dreams were crushed yet again, and I quickly realized that I actually had no idea what I was doing when it came to tackling this seemingly simple game.

Years later, I found myself facing the same exact dissapointment in college as a challenging problem came across me.[ I was / my friends and I were ]. The problem seemed vaguely familiar to me but I couldn’t quite get my finger on it. However, once I started working through [ it/ the problem ] in Java, insight eventually flooded through and when I finally got the program to work I realized it was the same Towers of Hanoi that had fascinated me as a child. Only now I was able to wrap my head around it and understand it at its most fundamental level.

That combination of thinking about a problem while tinkering with solutions in the form code, or “thinkering” as I call it is what I have found I live for. That “thinkering” is what set the gears of my mind turning and helped me develop a problem-solving mindset. That “thinkering” is what shifted my brain into overdrive to understand the complexity of the puzzle.

I found that/the brainstorming, drafting, writing, debugging, and analyzing process that went into creating this Java program was extremely rewarding and eye opening. Taking the rules and game mechanics of the Towers and reconstructing them in code from the ground up is what solidified the concept, and more importantly my passion, for computer science.

It was likee building the puzzle at the same time as solving it, and invited me into a way of thinking that forced me to consider the minute details and overall picture of the problem simultaneously.

Computer science is about the imaginative yet logical mindset that it compels me to adopt

- solve problems and create solutions

- develop a program from scratch

- create a program and solve the problems it presents

- some computer science filler thing to do

This mindset has helped me in myriad ways: from being able to analyze a question to find the best possible solution or approach, breaking down a problem to its most basic elements, or to even how to better efficiently handle my work and day to day life.

I would not have attained this mentality if it wasn’t for the ethos/ principles of computer science and the main reason why I will pursue it.

In addition to the mindset, another major reason why I gravitate towards computer science is the “creation” aspect of it. There is something about turning an idea into reality that

- motivates/ abstract word

I consistently ponder about how I can combine all the tools at my disposal to create something that could potentially change the world. It may sound cliché, but it really is the truth in a lot of ways. Much like a painter with his paintbrush, all a computer scientist needs is a computer to make their masterpiece. For me that could be my fashion app that I am currently developing that would help people categorize their wardrobe, build outfits, and plan out their fashion goals. For me that could be analyzing existing data about Chicago and gaining new insights on the average commute of CTA riders each day.

Going hand in hand with this, the impact and reach of the field is what also attracts me to it. I absolutely love the fact that the applications and products one can build with computer science has the power to touch/reach millions quickly/instantaneously and have a direct impact/effect on people’s lives.

- Can be used to have real impact

Lastly the versatility of computer science pulls me toward it. I view it as being “flexible” but in the sense that [later] down the line I can explore and contribute/synonym to other fields. I can go into the medical field and make a program that helps regulate a patient's breathing or conversely go into personal finance and help people budget and determine trends in their spending. In essence, it gives me the autonomy of choice.

I have met with computer scientist and people who work in tech to gain insight on the industry and the work that they do. This along with my own/independent research, has led to find many of the qualities that are universal among computer scientists are also present in me. Patience, determination, and deep critical thinking are some of the attributes I embody and take to heart on a daily basis. From their accounts, the responsibilities that they have are matters that I am genuinely interested in and see myself doing. I see myself building a website in which people can publish their artistic work, developing a platform in which families can sign up for social benefits, or even founding the next big social media network. Projects like these excite me and are spaces where I can utilize my skills to bring about positive change into the world.

In a field like computer science, I have learned from in the field that teamwork in the workplace is essential. Even though I get satisfaction from working independently, I really enjoy working with others and

- motivates me/ increases my work ethic.

My [ **passion/ passion and drive** ] [ for computer science ] makes me confident in the fact that not only am I right for computer science, but computer science is right for me.