While searching for a candidate to study and report on, I found myself stumped for choice. Ultimately, I feel that simply doing the job that you were hired to do isn't enough. Although solving problems that stretch the imagination is a major component of a computer scientist's job, the effect of causing great changes with incredible breakthroughs, like Alan Turing's work during World War II, is lessened somewhat by the knowledge that it was what they were hired to do.

I considered programmers who created great works on their own initiative, like Id Software, which was essentially comprised of a small group of people, working ridiculous hours to create their passion project. However, that route didn't end up going very far, as while their creation was remarkable, there wasn't much of their personal lives to reflect and write on. I could have picked one of the more unusual characters from the company, like John Romero, but I'd be hard pressed to say his work (and especially, his attitude) are worth celebrating to me.

To that extent, I wondered what I did consider worth celebrating, and what kind of computer scientist I wanted to become. I started at the very beginning of what we would consider computing and worked up until I landed on Charles Babbage. I had, of course, heard that Ada Lovelace, arguably the first programmer, based her work on his machines, so I read further and realised he was exactly what I was looking for. Before I get to why, though, I should stop fluffing up my introduction and establish just who exactly Charles Babbage was.

Born right at the end of the 18th century to an affluent family, his exact date and place of birth is of contention, not wholly unusual at the time. His father, Benjamin Babbage was an affluent banker, so Charles enjoyed a high-quality education. He would have enjoyed a carefree childhood, too, if he hadn't been so sickly. He made several attempts to study at country schools or grammar schools, but his poor health kept him frequently home-bound, with plentiful access to tutors. One of these was an Oxford tutor, who helped him achieve a high enough level of education to be admitted into Cambridge.

However, Babbage was well-read and self-taught on subjects like mathematics, especially algebra, which he loved, and he had achieved such a high level of study before arriving that he found his lectures to be dull and simplistic. He didn't like the university's approach to mathematics, which as Harvey W. Becher specifies in his reflection of Babbage's life in an article for the *Historia Mathematica,* was "trapped", entirely focusing on a "geometrical-physical approach to mathematics designed to prepare the student for reading Newton's Principia". As I understand it, this is the modern-day equivalent of studying only applied maths and the kind of mathematics used in physics rather than branching out into more abstract lines of thinking. For someone like Babbage, this simply wasn't enough.

This is where we begin the see why I chose Charles Babbage as my subject. It is, naturally, popular and fashionable to criticise the wealthy for their achievements. Surrounded by tutors, Babbage's success thus far can easily be attributed to the luck of his birth. As any psychologist will tell you, though, you must be willing to be helped before they can help you. Babbage could have easily skated by on life, retreated into his illness and made excuses

for his poor studies. He could have chosen the less imaginative path and followed in his father's footsteps, faded into anonymity by never learning or creating or teaching anything worth remarking on.

However, he seemed to be drawn by a genuine interest and he didn't just study deeply, he studied broadly. This is what appeals to me hugely as a computer scientist. While I think this applies to most carriers and all modes of study, the ability to branch out into ideas that seem distant from computing is highly prized and useful. After all, it's perfectly possible that a programmer assigned to design software for medical use needs a solid grasp of the biology they're working with. Or maybe they're asked to design a website or program a game, with which they'll need to understand more abstract concepts, like use of colour or shape in design or mood and pacing and movement in gaming. Computer Science may not move out of the STEM field much but expecting that a narrow knowledge of it, of only programming or only circuit design, will get a computer scientist anywhere is foolish, maybe even a little arrogant. This is something Babbage would have understood perfectly, his mode of study was mainly math-based, but he was an engineer and inventor also, he published works on politics and economy and even several, controversial works on theology, which he was passionate about.

Babbage's approach to fixing his problem with Cambridge, I find equally admirable. He set up the Analytical Society, since disbanded, whose primary aim was to promote the use of Gottfried Leibniz's method of differentiation in calculus over Newton's.

After graduation, he would go on to lecture to the Royal Institution on astronomy, but his teaching carrier never advanced much farther and he began work on the devices that would be his most influential work.

In Babbage's time, the most complex calculators were only able to perform basic arithmetic, their design influenced and frequently based off Gottfried Leibniz's Step Reckoner. However, Babbage was interested in a machine that could compute polynomial functions. He planned on using the method of finite differences to avoid the complication involved in using multiplication and division. He worked for around ten years between 1822 and 1831 with the engineer he hired to carry out his designs, Joseph Clement on the machine that would be known as the Difference Engine. However, it was never completed. Once Clement dropped out of the project, Babbage continued his work for at least a decade longer. Nearly twenty years later, in 1847, Babbage produced the designs for an improved version of the machine, "Difference Engine No. 2". However, because of his failings with the first Difference Engine, he was unable to receive funding for this new project, and the machine remained unbuilt. Recently, in 1991 his design was constructed and found to work perfectly, calculating results to 31 digits.

One incredibly good thing came about from Babbage's failures, however. As he worked on the Difference Engine, he began to work on the designs for an even more complex computer, which would be called the Analytical Engine. Unlike every machine before it, the Analytical Engine was a general-purpose computer, meaning it could be used for more than just one type of computation. It could be given data and run operations in sequence, had

memory and was able to print out its results. The machine used punch cards, inspired by the invention of Jacquard Loom, to control a mechanical calculator. The idea of a computer guiding itself through calculations was revolutionary and ahead of its time. However, like its predecessor, the Analytical Engine was never completed. Babbage would continue to work on it until his death in 1871.

His legacy wasn't wasted, however, as his work inspired one of the first ever programmers, Ada Lovelace. While in correspondence with Babbage, Lovelace developed an algorithm using the Analytical Engine to calculate a sequence of Bernoulli numbers. She was an advocate of Babbage's work and his invention and proved its reach and use.

Babbage's life wasn't particularly grand. He was sickly often and his pursuit of a career in lecturing left him in failure. His wealthy family kept him supported through his endeavours which were rarely successful. Regardless, I find him to be an extremely important and interesting person. He studied hard and broadly, enough to earn himself a place in one of the most prestigious and exclusive universities in the world. While there, he found the lectures lacking and, on his own initiative, took steps to improve them. He was a deeply and thoughtfully religious person and engaged in debate and wrote frequently on the subjects that interested him, broad and varied as they were. His life's work took him years of careful design and decades of dedicated construction. When it fell through, he continued to tinker and alter the design until he developed an even more revolutionary machine, one which would influence computers and their design going forward and inspire computer scientists from Lovelace onwards.

The mark he's left on the world seems inconspicuous, but I believe it was important, a chink in a chain that starts two thousand years ago and stretches forward to today.

References

Bbc.co.uk. (2018). *BBC - History - Charles Babbage*. [online] Available at: http://www.bbc.co.uk/history/historic_figures/babbage_charles.shtml [Accessed 25 Oct. 2018].

Cbi.umn.edu. (2018). *Charles Babbage Institute: Who Was Charles Babbage?*. [online] Available at: http://www.cbi.umn.edu/about/babbage.html [Accessed 25 Oct. 2018].

En.wikipedia.org. (2018). *Charles Babbage*. [online] Available at: https://en.wikipedia.org/wiki/Charles_Babbage#Natural_theology [Accessed 25 Oct. 2018].

En.wikipedia.org. (n.d.). *Jacquard loom*. [online] Available at: https://en.wikipedia.org/wiki/Jacquard_loom [Accessed 25 Oct. 2018].

W. Becher, H. (1980). Woodhouse, babbage, peacock, and modern Algebra. *Historia Mathematica*, 7(4).