In a 1995 interview 1995, Steve Jobs famously said: “Everybody should […] learn a computer language, because it teaches you how to think.” I could not agree with this more: software engineering is the most exciting thing I know. Having discovered it at the age of 10 by learning Pascal at my school’s extracurricular programming club, within just a few months I had rapidly progressed from the obligatory “Hello World” to writing my own version of the 1970s classic arcade game Space Invaders, complete with an adjustable shooting angle and custom game mechanics.

As my school in Saint Petersburg specialised in Maths and Computer Science, I had ample opportunity to hone my abilities. At first, I focused on learning the fundamentals of C; although my initial progress was slow and hard-won, I gradually came to terms with its syntactic rigour and continued to build on this foundation by studying object-oriented languages like C++ and Java. In consequence, I implemented an AVL tree from scratch, solidifying my grasp of recursion in the process, and used my new-found understanding of binary trees in Huffman coding. By using the OpengGL API I also designed and animated a functioning 3D vehicle to extend my knowledge to other CS fields.

Coming to the UK last year to continue my education had allowed me to focus on the subjects relevant to my future career choice. A-levels in Maths and Further Maths gave me a new perspective on familiar concepts, such as learning the relevance of probability distributions and statistical expectations to data compression. Teaching myself modules on Decision Maths, I discovered and implemented Graph Theory algorithms such as Prim's and Kruskal's in Python to reinforce my practical knowledge. Finally, I was able to write dynamic simulations of global economic markets by combining my knowledge of mathematical models with A-Level Economics.

Like every great technology, computers give their users unique ways to misuse them. Watching the documentary The Social Dilemma made me consider the downsides of an interconnected global society, which allows big tech companies to collect and trade personal data to shape public opinions and make more profit. To investigate further, I read Free Software, Free Society by Richard M. Stallman, which introduced me to the world of Free and Open-Source Software. This became the topic of my EPQ, Linux: An OS You Control, which explored how, by meticulously decomposing every action undertaken in the OS, a user can control the system, instead of the system controlling the user.

I always strive to grow my knowledge of coding and the algorithms involved therein. Behind Deep Blue, a book about the AI that defeated the world chess champion in 1997, encouraged me to enrol into a two-week summer course on AI at the University of Oxford, where I studied different types of machine learning and the fundamentals of constructing a deep neural network. Clean Code by Robert C. Martin and Andrew Hunt's The Pragmatic Programmer have given me valuable practical advice, improving the quality of my code and making it more efficient. Last summer, a two-week internship in a Swiss digital currency investing firm Euratek GmbH, showed me the challenges of dealing with the inherently volatile cryptocurrency markets, and the importance of using effective software solutions to mitigate their unpredictability, giving me invaluable experience as a result.

I believe that physical and mental health are intertwined, and as a result have played tennis for 7 years and have 4 years of MMA experience. I enjoy volunteering, having recently coordinated a Night Walk for the National Autistic Society, and was recently appointed to the position of Digital Innovator on the Student Council. This has given me a great opportunity to help out the college by, for example, administrating and improving a local rewards system "Epraise".