

# Oliver Cruickshank

## Curriculum Vitae

### PERSONAL DETAILS

---

Birthdate: December 11th 1999  
Address: 39 St. Bartholomews Court, Riverside, Cambridge  
Phone: (+44) 07472007901  
Email: oliverlcruickshank@gmail.com  
Github: www.github.com/oliverlars

### PERSONAL STATEMENT

---

I'm a first year undergraduate studying Computer Science at the University of York. I have a deep interest in low-level programming, including topics such as: compilers, virtual machines, programming micro-controllers and performant graphics. I am hoping to secure an internship this summer to further my interest in compilers and low-level programming to have the opportunity to learn from engineers with more experience and knowledge.

### EDUCATION

---

**Degree: University of York: Computer Science** 2018-2021

**A levels: Hills Road Sixth Form Centre** 2016-2018  
Achieved grades: Maths B, Physics B, Computing B, Extended Project B

**GCSE: St Bede's Inter-Church School** 2011-2016  
Achieved grades: A A A A A A A A B

### PROJECTS

---

**BMO, a language to typeset documents in C++** 2019  
I designed and built a language for typesetting documents. It is similar to groff, producing PDFs as output that are nicely formatted. My CV was formatted and rendered using BMO

**Picoblaze 6 compiler and virtual machine in C** 2019  
The Picoblaze is a soft-core CPU designed to run on Xilinx FPGAs. I wrote a compiler that generates bytecode according to their specification and then implemented a virtual machine that runs the bytecode

**Fast Embree based Pathtracer in C** 2018  
Using Intel's Embree framework for high performance ray intersection tests to make a pathtracer. I wrote a simple threadpool to do tile-based rendering, a fast OBJ loader as well as progressive rendering using Win32

**Minecraft-like world generator in C++** 2018  
Using OpenGL, I made a minecraft-like world which uses perlin noise to generate landscapes

**Pathtracer from scratch for my Computer Science coursework in C++** 2017  
I wrote a pathtracer from scratch which used a KD-tree for fast search for possible triangle intersections among common features like depth of field, interpolated normals, and textures

### SKILLS

---

Languages: C, C++, Python, Java, Pascal  
Technologies: OpenGL, Embree, SIMD