

# Curriculum Vitae

James Hobson (28/01/1999)

April 7, 2021

## Summary

I currently working in the software R&D department at Jotron AS. Although this is my first post-university job, I have a lot of experience developing software in industry. I have been programming in various languages since the age of 9, being proficient in a large number of tools and languages. My main interests have always been where theoretical computer science meets the real world. This includes compiler design, formal verification, correctness proving, and using abstract mathematics to deliver high performance without sacrificing safety or security. Outside of work, I contribute to open source projects and am jazz/funk bass player.

## Skill Highlight

- Proficiency in C/C++, Haskell, Scala, C#, F#, OCAML, BASH, JS, Python, Assembler (ARM and X86\_64)
- Ability to quickly understand large, complex systems
- Proficiency in Linux and other UNIX Systems
- Knowledge of LaTeX and experience using it for technical writing and academic
- Proficiency in applying formal methods and abstract maths to code writing
- Leadership and organisation - I ran the Oriel College philosophy society and was manager of the group project in Oxford

## Education

Uppingham School: 2012-2015. (UK) Abbey Grange Academy: 2015-2017. Oxford University: 2017-2020

- Oxford University on computer science and Philosophy: 2:1
- **A-Levels:** Maths: A\*, Physics: A, Economics: A, Further Maths: B
- **AS-Levels:** Further Maths: A, Physics: A, Economics: A, General Studies: A, Maths: B
- **(I)GCSEs:** 9 A\*s, 1 As and 2 Bs – A\*s include Maths and Physics. Also Further maths additional qualification level 3: A (highest possible grade)

My University Modules:

Subject	1st Year	2nd Year	3rd Year
Computer Science	Functional Programming, Imperative and OOP Programming, Design and Analysis of Algorithms, Discrete Maths, Probability	Compilers, Models of Computation, Lambda Calculus, Algorithms 2, Concurrent Programming Computer Security	Formal Verification, Principles of Programming Languages, Quantum Information, Computer Architecture 2
Philosophy	Into to Logic, Elements of Deductive Logic, General Philosophy Analytic and Historical, Turing on Computability & Intelligence	Philosophic Logic, Philosophy of Logic and Language	Thesis on Logical Consequence and Completeness of Formalist Systems

## Experience

### Jotron (Norway) Summer 2020 - Present

- Researching and developing type safe configuration tools with mathematical guarantees with respect to safety and security
- Started and is leading Jotron's move into open source software
- Continued development of the project that I began last year (lead developer)
- Taken over lead for linux devops

### Jotron (Norway) 12-Week Contract – Summer 2019

- I was set the task to build a screen recorder fit for air traffic control, an environment where safety, data integrity and performance were paramount
- I was the only developer on the project, having control over the entire code base.
- The solution used abstract algebra to model data pipelines that could be rearranged and simplified to achieve maximum performance whilst guaranteeing that optimisations wouldn't result in unintended behaviour
- Jotron have since hired me as a consultant for R&D

### ARM (UK) 12-Week Internship – Summer 2018

- Built a project in C (and some Haskell) to do data capture and analysis as part of the Mali Display test framework
- Independent project with a set end goal but no information past the specification and only me as a developer
- Experienced working on a semi-research project, where the final product matters as much as the proof of concept
- Built tools for specific data analysis (which I have been told were actually useful!)
- Experienced getting up to speed with technology I had not even heard of before

## Backend Web Development Summer Job at Enjoy Digital (UK) – Summer 2017

- Building CMSs that could be used by clients to customise and create pages with no risk of breaking their webpage
- Elastic Search integration for websites – extracting key data from multiple sources and arranging it in complex data types that can be queried in flexible ways (No-SQL)
- Independent research project into lightweight machine learning for film recommendations
- Independent research project into population modeling to optimise website cache
- Wrote several tools still used by the business today

## Game Development Uppingham Summer School (UK) – one week each summer 2013-2015

- Teaching assistant on a week-long summer school for people aged 11-18 on game development using Unity, Blender and other software
- Worked in C#, building teaching materials that students could use in their games: generic scripts that they could easily modify to learn the thought behind algorithm design without needing much C# knowledge
- Experience in communicating technical ideas clearly, as many of the students had only little technical experience
- Experience working with people with various disabilities
- Experience with tight deadlines (one week to build many games)

## Personal Projects

Ever since I started to code, I've been building small projects. Here are some highlights:

- Active member of the openSUSE linux project
- Contributor to the linux appimage project
- Currently creating a youtube tutorial series on the haskell programming language
- CTL Model Checker written in Haskell
- A Distributed Computing Platform where phones solve small parts of large problems when charging at night (Won Oxford University Prize)
- Compiler written in Haskell to compile contracts with a web form as input
- DIY Smart Lock for room in college (c)
- Set cryptographic challenges in the College Newspaper (mostly Haskell, BASH, JS, HTML)
- Simple Harmonic Oscillation analyser for A2 Physics experiment using OpenCV (Win32 C++, GNUPlot)
- Machine Learning Smell Detector for class rooms (Qt C++/C, Wiring  $\pi$ )
- Exoplanet Detection Emulator (Qt C++)
- Made C programming tutorials for RISC OS (to be found on Youtube)
- Designed 3 simple microprocessors in Logisim (drag and drop logic gates to build reusable components)
- Multiple Arduino and Raspberry  $\pi$  projects
- Implemented many graph algorithms that I learnt in D1 A Level Maths in Qt C++