joshua.blinkhorn@
uni-jena.de
(+49) 1535 9722400

Programming

C ○
C++ ○
Python ○
Javascript ○
PHP ○
SuperCollider ○

Scripting

HTML ○
CSS ○
LaTeX ○

Platforms

Arch Linux ○
Mac OS ○
Windows ○
Git / GitHub ○

Chess

Stockfish ○
UCI ○
Python-chess ○
HIARCS ○

Languages

English (native) ○
German (B1.2) ○

Joshua Blinkhorn

Postdoctoral Researcher in Artificial Intelligence

Profile I am a self-motivated, hard-working and passionate individual. Since 2015 I have conducted research in *proof complexity*, at the intersection of logic, artificial intelligence and computational complexity. I specialise in *quantified Boolean formulas*, which express winning strategies in two-player games like chess and Go. In the past, I have worked as a musician and sound recordist. I love to read and to create. I've been a lifelong chess fan, and I'm thrilled to see how the online game is thriving.

Chess Training Software

This summer, in a drive to improve my chess, I began developing a suite of command-line chess training tools in Python. The first installment is an opening trainer based on *spaced repetition* and *active recall*, efficient memorisation techniques pioneered by the flashcard system Anki. The next installment, an endgame trainer, is a much bigger challenge. A given endgame technique may be executable in various ways – how can we automate the teaching of chess *ideas*, rather than memorising lines?

Recent Timeline

Since December 2019 – University of Jena, Germany
I currently hold a research position in the Computer Science Institute.

September 2015 - December 2019 – University of Leeds, UK
I obtained a PhD. in proof complexity from the School of Computing.

September 2008 - June 2015 – Open University, UK
I obtained a first-class Batchelor of Arts in Mathematics, with average marks of 98% (coursework) and 96% (examinations).

Teaching

Summer 2020 - Cryptology
I directed students in the implementation of cryptosystems and cryptanalysis in C++ and Python.

Summer 2019 - SAT Solving
I oversaw students' development of satisfiability solving tools in a language of their choice, predominantly C++, Python and Java.

Autumn 2015 & 2017 - Procedural Programming
I worked as lab session demonstrator and coursework marker in this introductory course on the C language.

Publications

I have co-authored nine conference publications at six computer science venues, including the *International Joint Conference on Artificial Intelligence* and *Logic in Computer Science*, both of which have the top CORE ranking 'A*'. In 2017, my article won the best paper award at the *International Conference on Theory and Applications of Satisfiability Solving*, which has CORE ranking 'A'. I have also co-authored four journal publications. For more details, see my DBLP list or my academic CV.