

ZK Golf: Designing a Course of Constraint-Optimization Challenges

Advisor: **Florian Lugstein**

Motivation

Zero-knowledge proofs (ZKPs) such as SNARKs let you prove statements about secret data without revealing the secrets. Before a computation can be proven, it has to be expressed as an arithmetic circuit, typically in the form of a *Rank-1 Constraint System* (R1CS). The number of constraints directly determines how expensive it is to generate a proof, so writing *small* circuits is key for building efficient ZK applications.

Like in code golf, the goal is to express a function in as few constraints as possible, using tricks such as folding linear operations, hinting non-deterministic witnesses, or clever bit manipulations. However, cutting one constraint too many can leave the circuit *under-constrained*, allowing a cheating prover to prove false statements.

In this project, you will design a **ZK Golf course**: a CTF-style set of challenges ("holes") of increasing difficulty that teaches how to write efficient and secure circuits.

Goals and Tasks

- Understand R1CS and learn common constraint optimization techniques
- Build a set of challenges with reference solutions and an automated checker for correctness, soundness, and constraint count
- Analyze optimization strategies and pitfalls leading to under-constrained circuits



Hole 3: $y = 1$ **if** $x = 0$**, else** $y = 0$

✔ **Par (2 constraints)**
$x \cdot inv = 1 - y$, $x \cdot y = 0$

✘ **Hole-in-one? (1 constraint)**
$x \cdot y = 0$ also accepts x=0, y=0

Fewest constraints wins, but only if the circuit stays sound.

Literature

- J. Thaler
  Proofs, Arguments, and Zero-Knowledge
  FnT Privacy and Security 2022
  https://people.cs.georgetown.edu/jthaler/ProofsArgsAndZK.html
- S. Chaliasos et al.
  SoK: What Don't We Know? Understanding Security Vulnerabilities in SNARKs
  USENIX Security 24
  https://eprint.iacr.org/2024/514

Courses & Deliverables

- ☑ **Introduction to Scientific Working**
  Short report on background
  Short presentation
- ☑ **Bachelor Project**
  Project code and documentation
- ☑ **Bachelor's Thesis**
  Project code
  Thesis
  Final presentation

Recommended if you're studying

☑ CS ☑ ICE ☑ SEM

Prerequisites

- Interest in ZKPs, algebra, and puzzles
- Programming (Rust)

Advisor Contact

florian.lugstein@tugraz.at