# Sacha Ben-Arous

sacha.ben-arous@ens-paris-saclay.fr | +33 644 39 10 99

# **Education**

Lycée Janson de Sailly, Classes préparatoires aux grandes écoles (CPGE), Paris. Sept 2020 – July 2022

• MPSI, MP\*: Mathematics, Physics, Computer Science.

**École Normale Supérieure Paris-Saclay**, Department of Computer Science. Sept 2022 – July 2023

• License 3: Double License Informatique, University Paris-Saclay.

**École Normale Supérieure Paris-Saclay**, Department of Mathematics. Sept 2023 – July 2026

• License 3 : Double License Mathématiques, University Paris-Saclay.

• Master 1: Master Jacques Hadamard, University Paris-Saclay.

# **Interships**

The MP-LWE problem, supervised by Alice Pellet-Mary, University of Bordeaux.

June 2022 - July 2022

• Study of the Learning With Errors (LWE) problem and reduction of some of its polynomial variants.

Paradifferential K.A.M theory, supervised by Thomas Alazard, E.N.S Paris-Saclay.

April 2023 – June 2023

• Simplified approach of K.A.M-like theorems using the theory of paradifferential operators.

## **Projects**

#### Go language compiler

- Developed a compiler of a simplified version of the Go language to ASMx86-64.
- Tools Used: OCaml, ASMx86-64, Menhir, Yacc.

# Automatic proofs of first order predicates

- Implemented the method of semantic tableaux to constructively prove or refute a first order predicate.
- Tools Used: OCaml, Menhir, Yacc.

### Minimal computer

- Emulated, in a custom framework, a simple computer with a cpu and ram, only using logic gates.
- Tools Used: Assembly x86-64.

#### **Custom shell**

- Wrote my own shell for Unix OS.
- Tools Used: C, Menhir, Yacc.

#### Syntaxic analyser

- Wrote a syntaxic analyser for a toy language that checks if a program is syntactically correct, then pretty print it and performs random specification tests.
- Tools Used: C, Flex.

#### **Awards**

**Mathematics Olympiads, Martinique (1st):** Ranked 1st among high school students in the department of Martinique, France.

# **Technologies**

Languages: Python, MT<sub>F</sub>X, Sagemath, OCaml, C, Assembly x86-64.