Tom Cornebize

Student in computer science

+33(0)6 74 69 59 41
tom.cornebize@ens-lyon.org
cornebize.net

Education

Ensimag — **Grenoble, France**
Grenoble INP - Superior National School of Applied Mathematics and Computer Science — *2015–2017*
Graduate specialization in Information Systems Engineering.

ENS Lyon — **Lyon, France**
École Normale Supérieure de Lyon — *2013–2015*
Undergraduate and graduate intensive program in theoretical computer science.
Obtained a Bachelor of Science, with great honor.

Université Joseph Fourier — **Grenoble, France**
Grenoble University — *2011–2013*
Undergraduate program in computer science and mathematics.

Experience

Research internships

Bull, HPC R&D — **Grenoble, France**
High performance computing : job isolation in fat tree topologies. — *May 2015–August 2015*
Under the supervision of Matthieu Perotin.
- Designed several algorithms to prevent the leak of sensible information in a cluster.
- Implemented a proof of concept, in Python, to obtain experimental results.
- Implemented system integration, in Python.

Inria, AOSTE team — **Sophia-Antipolis, France**
Modelisation and verification of concurrent systems. — *June 2014–July 2014*
Under the supervision of Robert de Simone.
- Studied classical models (Büchi automata, Petri nets, temporal logic, synchronous languages).

Verimag laboratory — **Grenoble, France**
Monitoring of distributed systems — *June 2013–July 2013*
Under the supervision of Yliès Falcone.
- Designed an algorithm for decentralized monitoring of distributed systems.
- Implemented a proof of concept, in OCaml, to obtain experimental results.
- Published a report at FORTE 2014 : "Efficient and Generalized Decentralized Monitoring of Regular Languages".

Laboratoire d'informatique de Grenoble (LIG) — **Grenoble, France**
Monitoring of distributed systems — *June 2012*
Under the supervision of Yliès Falcone.
- Performed experimentations and proposed several optimizations.

Software projects

Platypus
- A modular and open source question answering framework. Team of seven students.
- Developed a question parsing module in Python, with a grammatical approach (Stanford CoreNLP and NLTK libraries).

SAT solver
- Developed a program to solve the SAT problem, based on the DPLL algorithm, in C++ language.
- *Watched literals* and *clause learning* heuristics.

Cellular automata, in C. Used *MPI*.
Simulation of distributed search, in Erlang.
P2P client, in C. Used *pthread* and *socket*.

Languages

French: Mother tongue
English: Fluent
German: Basic

Computer skills

Programming languages: Python, C, C++, OCaml, Erlang, assembly languages (ARM and MIPS)
Presentation languages: LaTeX, Markdown
Distributed systems / parallel programming: MPI, pthread, socket
Miscellaneous: GNU/Linux, Git, unit testing