

Nathanaëlle Courant

Étudiant – École Normale Supérieure

Formation

- 2018–2019 **Stage de recherche, Équipe Celtique, Inria Rennes.**
Stage de recherche avec Thomas Jensen et Alan Schmitt : « A Rule-based Format for Operational Semantics : formalisation and applications ».
- 2016–2018 **Master Parisien de Recherche en Informatique (MPRI), École Normale Supérieure, Paris.**
Stage de recherche de M1 avec Natarajan Shankar ; stage de recherche de M2 avec Xavier Leroy. Admis mention très bien, moyenne 19,46/20, rang 1^{er}.
- 2015–2016 **License 3, École Normale Supérieure, Paris.**
Double-licence mathématiques et informatique. Stage de recherche avec Caterina Urban.
- 2013–2015 **Classes préparatoires, Lycée du Parc, Lyon.**
- 2013 **Baccalauréat série S, Lycée la Martinière Monplaisir, Lyon.**
Mention très bien.

Publications

- 2017 **TACAS 2017, Nathanaëlle Courant, Caterina Urban.**
Precise Widening Operators for Proving Termination by Abstract Interpretation

Expérience d'enseignement

- 2018–2019 **Colles en mathématiques en MP*.**
Une heure de colle par semaine en mathématiques, dans la classe de MP* du lycée Chateaubriand (Rennes). Également quelques heures de préparations aux oraux en mathématiques et en informatique à la fine de l'année.
- 2014–2018 **Cours au club de mathématiques discrètes de Lyon.**
Club pour lycéens pour la préparation aux olympiades internationales de mathématiques. Environ un cours d'une journée par an.
- 2015–2016 **Cours de soutien en mathématiques en MPSI.**
Petit cours donnés à un élève de MPSI, 2h/semaine pendant environ 6 mois.

Résultats

- 2015–2017 **ACM-ICPC SWERC (Southwestern Europe Regional Contest).**
Équipe "ENS Ulm 1" ; classés 3^e en 2015, classés 1^{ers} et qualifiés pour la finale mondiale 2017 en 2016 (classés 34^e), classés 1^{ers} et qualifiés pour la finale mondiale 2018 en 2017 (classés 56^e).

45 rue d'Ulm – 75005 Paris – FRANCE

☎ +33 7 81 62 69 42 • ✉ nathanael.courant@ens.fr

- 2016 **Google Code Jam.**
Qualifié pour la finale, classé 18^e.
- 2016 **Google Hash Code.**
Équipe “OCaml4Ever” ; classés 9^e.
- 2015 **Concours X-ENS.**
Classé 2^e à l'ENS Paris, 1^{er} à l'ENS Lyon et l'ENS Rennes, 6^e à l'ENS Cachan, et 7^e à Polytechnique.
- 2014–2016 **Prologin.**
Classé 1^{er} en 2016, 10^e en 2015, et 4^e en 2014.
- 2013 **Olympiades internationales de mathématiques.**
Membre de l'équipe de France. Médaillé d'argent.
- 2013 **Concours général des lycées.**
1^{er} prix en physique et en sciences de l'ingénieur, 2^e prix en mathématiques.

Compétences

- Langues Français (langue maternelle), Anglais (courant), Allemand (intermédiaire), Espéranto (débutant).
- Langages de programmation *Par ordre de maîtrise décroissant* : Python, OCaml, Coq, Lua, C/C++, un peu d'expérience avec d'autres langages dont : Scheme, Assembleur x86-64 et ARM, Forth.

Compétences théoriques

- Informatique Compilation, sémantique, typage, λ -calcul, logique linéaire, interprétation abstraite, algorithmes, calculabilité, complexité, théorie des catégories, apprentissage statistique, optimisation convexe, cryptographie.
- Mathématiques Théorie de la mesure, algèbre de base, topologie, analyse complexe, probabilités.

Expérience de programmation

Projets
principaux

- Un générateur de code pour le modèle polyédrique, qui génère un arbre de syntaxe pour le cas de plusieurs polyèdres, écrit et formellement vérifié en Coq.
- Résolution des 677 problèmes de projet Euler (au moment de l'écriture de ceci).
- Un compilateur d'un petit langage purement fonctionnel vers C, incluant des types de données algébriques, des effets algébriques avec continuations *multi-shot*, un GC, et du typage statique qui vérifie les effets.
- Un générateur de code, d'un petit langage fonctionnel vers un langage impératif, incluant la preuve de correction du comptage de références et des mises à jour destructives, écrit et formellement vérifié en PVS.
- Un compilateur d'un sous-ensemble de Scala vers de l'assembleur x86-64, écrit en OCaml.
- Un compilateur de *netlists* vers C écrit en OCaml, et un CPU écrit dans ce langage de *netlists*, généré par un programme en Python.
- Un système d'exploitation minimaliste pour Raspberry Pi, écrit en C, C++ et assembleur ARM.
- Un système Forth, respectant le standard Forth-83, écrit en Python et Forth.
- La vérification formelle du théorème de réciprocité quadratique et du théorème de Baire en Coq.
- *Core developer* de Minetest, un jeu vidéo open-source.

Autres projets Autres projets de cours, programmation de petits jeux vidéos au collège/lycée, etc.