CS-628 **Interactive Theorem Proving CS**

Barrière Aurèle, Pit-Claudel Clément

Cursus	Sem.	Type
Computer and Communication Sciences		Opt.

Language of teaching	English
Credits	6
Session	
Exam	During the semester
Workload	180h
Hours	**6**
Lecture	2
Exercises	1
Practical work	3
Number of positions	

Frequency

Every year

Remark

Spring 2024

Summary

A hands-on introduction to interactive theorem proving, proofs as programs, dependent types, and to the Coq proof assistant. Come learn how to write bug-free code!

Content

Draft syllabus
- Intro to the Coq proof assistant (logic, higher-order functions, tactics)
- Functional programming (inductive types and fixpoints)
- Structural induction (data structures and verified algorithms)
- Interpreter-based program semantics (intro to compiler verification)
- Inductive relations (predicates, rule induction)
- Automation and tactics I (bottom-up reasoning and logic programming)
- Operational program semantics (small- and big-step semantics)
- Program logics (hoare triples)
- Automation and tactics II (top-down reasoning)
- Type systems (Simply-typed lambda calculus)
- Dependent types and equality proofs
- Automation and tactics III (proofs by reflection)
- Real-world interactive theorem proving (guest lecture)

Teaching methods: (e.g., ex cathedra, in-lab project, field trip)
- Lectures
- Live-coding sessions

Expected student activities:
- Weekly programming and verification assignments

Assessment methods:
- Take-home programming and verification assignments

Note

Learning outcomes:
Implement purely-functional algorithms in the Gallina language; Translate informal requirements about software into precise mathematical properties; Plan and carry out mechanized proofs in Coq (e g maths algorithms compilers type systems); Automate repetitive proof tasks

Keywords

Interactive theorem proving, verification, intuitionistic logic, program proofs, functional programming

Learning Prerequisites

Recommended courses

- Formal Verification (CS-550)
- Computer language processing (CS-320)
- Foundations of software (CS-452)

Resources

Bibliography

- https://softwarefoundations.cis.upenn.edu/
- http://adam.chlipala.net/frap/
- https://coq.inria.fr/distrib/current/refman/

Moodle Link

- https://go.epfl.ch/CS-628