

1st Workshop on Computer Algebra in the Age of Types

Computer algebra, being one of the oldest computational disciplines, keeps growing rapidly. Most code is available as open source software: Constructive code tends to be written in very weakly typed systems like Python (notably Sage), GAP, C, or C++. Formal systems naturally carry strong typing and formal proof tools.

There is still, somewhat surprisingly, very little connection between the two disciplines, though many attempts have been made to bridge the gap by devising new systems that incorporate strong typing, proof assistants, and powerful (bespoke) programming languages.

This workshop advocates a slightly different approach: For modern computational mathematics to thrive we want to promote the use of types to

- compose existing constructive and formal systems with a low initial overhead, and a strong incentive towards eventual formal checkability and correctness,
- enable development of domain specific tools that make computations and algorithms in mathematical domains more natural, concise, and avoid large classes of common errors and cognitive overhead.
- provide access to machine verified proofs for use in computation and algorithm development
- provide access to efficient computations for use in proofs
- enable effective automated testing a-la Haskell's Quickcheck

This workshop provides an incubator for this approach, featuring a tutorial on the programming language Idris (<https://www.idris-lang.org/>) by its author Edwin Brady, and an introduction to the system MMT (<https://uniformal.github.io>), inviting practising researchers in computer algebra, formal methods, and programming languages to present and discuss approaches and form a common vision for the next generation of computer algebra.



This workshop is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 676541.

Copyright © by the paper's authors. Copying permitted for private and academic purposes.

In: O. Hasan, M. Pfeiffer, G. D. Reis (eds.): Proceedings of the Workshop Computer Algebra in the Age of Types, Hagenberg, Austria, 17-Aug-2018, published at <http://ceur-ws.org>