PoP Seminar

Verifying Concurrent C Programs with Concurrent Separation Logic

William Mansky

Abstract:

In this talk, I will present my work on reasoning about concurrent programs with the Verified Software Toolchain (VST). Separation logic extends Hoare logic with ideas of memory and ownership, used to model the behavior of heap-manipulating programs. Permissions on memory are a natural way of thinking about both sequential and concurrent programs, but concurrency also brings its own challenges: how do threads communicate? Who owns a shared data structure? How can we account for relaxed memory models and low-level atomic operations? Using ideas from the newest generation of concurrent separation logics, we can come up with consistent reasoning principles for both lock-based and lock-free programs, and prove the correctness of non-blocking communication protocols and data structures. These proofs are both formalized in Coq and connected to working C code.

Bio:

William Mansky is a postdoc at Princeton University working with Andrew Appel on verifying concurrent C programs. He received his PhD from the University of Illinois at Urbana-Champaign under Elsa Gunter, and spent two years working with Steve Zdancewic on the Verified LLVM (Vellvm) project. His research interests include interactive theorem proving, program semantics and correctness, compiler correctness, and concurrency.

Monday, March 20, 2017
Gates Hillman Center 9115
3:30 PM – 4:30 PM