

# Bringing Techniques from Software Engineering into Scientific Software

**Eric L. Seidel, UCSD**  
Gabrielle Allen, U. Illinois

# Background & Motivation

- 5th year PhD student in Programming Languages
- Worked with Cactus team (HPC framework for scientific apps) during undergrad
- Interested to attend WSSSPE to understand if my, or related, research in computer science can be applied to research software.

# Correctness

- Starts from high-level **specification**
  - Functional Correctness
  - Memory Safety
  - Deadlock Freedom

# Ensuring Correctness

- **Prove** code correct
- **Test** code against broad range of (automatically generated) inputs
- **Synthesize** code from specification

# Technical Challenges

- Proofs
  - Semi-manual process, often requires understanding of program logic
  - Cryptic error messages, not enough context
  - Floating-point numbers
- Testing
  - Ensuring adequate coverage (scaling issues)
  - Providing oracles for outputs
- Please share experiences with verification tools!

# Social Challenges

- Collaboration between PL/SE and Research Software communities
  - Opportunities for internships?
  - How to reward students for participating?
- Incentives to produce **usable** tools in addition to novel techniques
  - Research Software community could help make case to funding agencies?