

# SHANGYIN TAN

(+1) 765 - 427 - 2861 ♦ [tan279@purdue.edu](mailto:tan279@purdue.edu) ♦ <https://shangyit.me>

## EDUCATION

---

### Purdue University

*Bachelor of Science in Computer Science Honors*

2018 - 2022

*West Lafayette, US*

- GPA: 3.98/4.0, Major GPA: 4.0
- Corporate Partner Scholarship
- PurPL Undergraduate Researcher
- Graduate Courses: *Algorithms, Programming Languages, Program Reasoning*

## RECENT PROJECTS

---

### Compiling Symbolic Execution

*Undergraduate Research* (advised by Guannan Wei and Tiark Rompf )

May 2020 - Present

*West Lafayette, US*

- <https://github.com/Kraks/sai>
- Compile efficient symbolic executions via multi-stage programming
- Build backend to generate SMT solver calls
- Lead the development of multiple *LLVM* symbolic execution compilers
- Publications: [OOPSLA 20], [ESEC/FSE 21], [PEPM 22]
- Submissions under review: [USENIX Security 22]

### Data-driven Inductive Invariants Inference

*Honors Research* (advised by Benjamin Delaware)

Sep 2021 - Present

*West Lafayette, US*

- Infer inductive invariants for recursive client programs

### Interactive Program Synthesis for TensorFlow

*Undergraduate Research* (advised by Tianyi Zhang)

July 2021 - Present

*West Lafayette, US*

- Design interactive interface for TensorFlow operation synthesis
- Create tutorial and conduct user-studies

### W<sup>2</sup>: Synthesising Responsive Webpage from Wireframe

*Coruse Project* (advised by Roopsha Samanta)

March 2020 - Aug 2020

*West Lafayette, US*

- <https://github.com/TigerHix/W2>
- Design an algorithm to infer hierarchical layout from static structure
- Transform static graph to responsive webpage (HTML)

### MiniScala: a Small Scala Compiler

*Course Project*

Jan 2020 - May 2020

*West Lafayette, US*

- Parse and compile *Scala* source code to X86-64 assembly
- Infer and check types of the input program
- Optimize via Dead Code Elimination, Constant Folding, CPS Transformation, etc

## PAPERS UNDER REVIEW

---

1. [USENIX Security 22] Shangyin Tan, Guannan Wei, and Tiark Rompf. The essence of compiling symbolic execution. In *USENIX Security Symposium*. USENIX Association, 2022

## PUBLICATIONS

---

1. [PEPM 22] **Shangyin Tan**, Guannan Wei, and Tiark Rompf. Towards partially evaluating symbolic interpreters for all (short paper). In *PEPM@POPL*. ACM, 2022
2. [ESEC/FSE 21] Guannan Wei, **Shangyin Tan**, Oliver Bracevac, and Tiark Rompf. LLSC: a parallel symbolic execution compiler for LLVM IR. In *ESEC/SIGSOFT FSE*, pages 1495–1499. ACM, 2021
3. [OOPSLA 20] Guannan Wei, Oliver Bracevac, **Shangyin Tan**, and Tiark Rompf. Compiling symbolic execution with staging and algebraic effects. *Proc. ACM Program. Lang.*, 4(OOPSLA):164:1–164:33, 2020

## PRESENTATIONS

---

1. **SPLASH 2021 SIGPLAN Papers Track**  
*Compiling Symbolic Execution with Staging and Algebraic Effects* Oct 2021
2. **PurPL Reading Group**  
*Data types a la carte* Aug 2020

## EXPERIENCES

---

### Student Volunteer

- SPLASH 2020: Review talk videos. Monitor Q&A sessions.
- SPLASH 2021: Coordinate hybrid video and streaming devices

### Undergraduate Teaching Assistant

*Discrete Math, System Programming, Algorithms Analysis, ...*

Jan 2019 - Jan 2021

*West Lafayette, US*

- Conduct recitations to help students with problem solving
- Advise students in lab debugging
- Monitor online Q&A forums like Piazza

### Selected Coding Contests

*Higher Ranked Participant*

2018 - 2020

*Midwest, US*

- 3<sup>rd</sup> in Tech Challenge Google 2019, Chicago
- 2<sup>nd</sup> in Sandia Coding Challenge 2018, West Lafayette

## SKILLS

---

### Familiar with

C, Scala, Python, C++

### Have worked with

Haskell, Coq, X86-64, Java, Javascript, Scheme, L<sup>A</sup>T<sub>E</sub>X, LLVM, MatLab

### Tools

GDB, Git, QuickCheck, SAT/SMT solvers (Minisat, STP, Z3)

(Skills in the same row are in random order)