

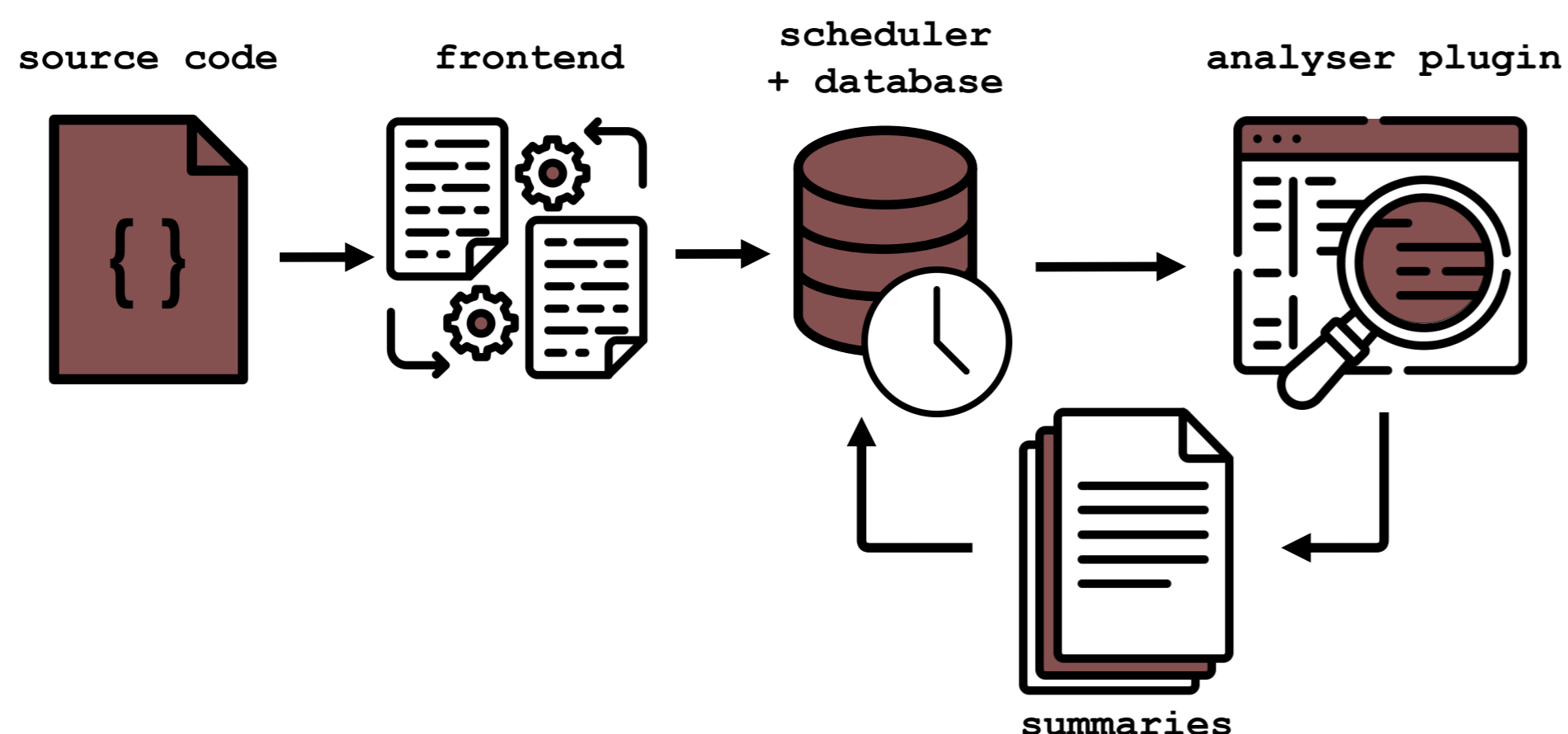
STATIC DATA RACE DETECTION IN LOW-LEVEL C CODE

Author: Lucie Svobodová

Supervisor: prof. Ing. Tomáš Vojnar, Ph.D.



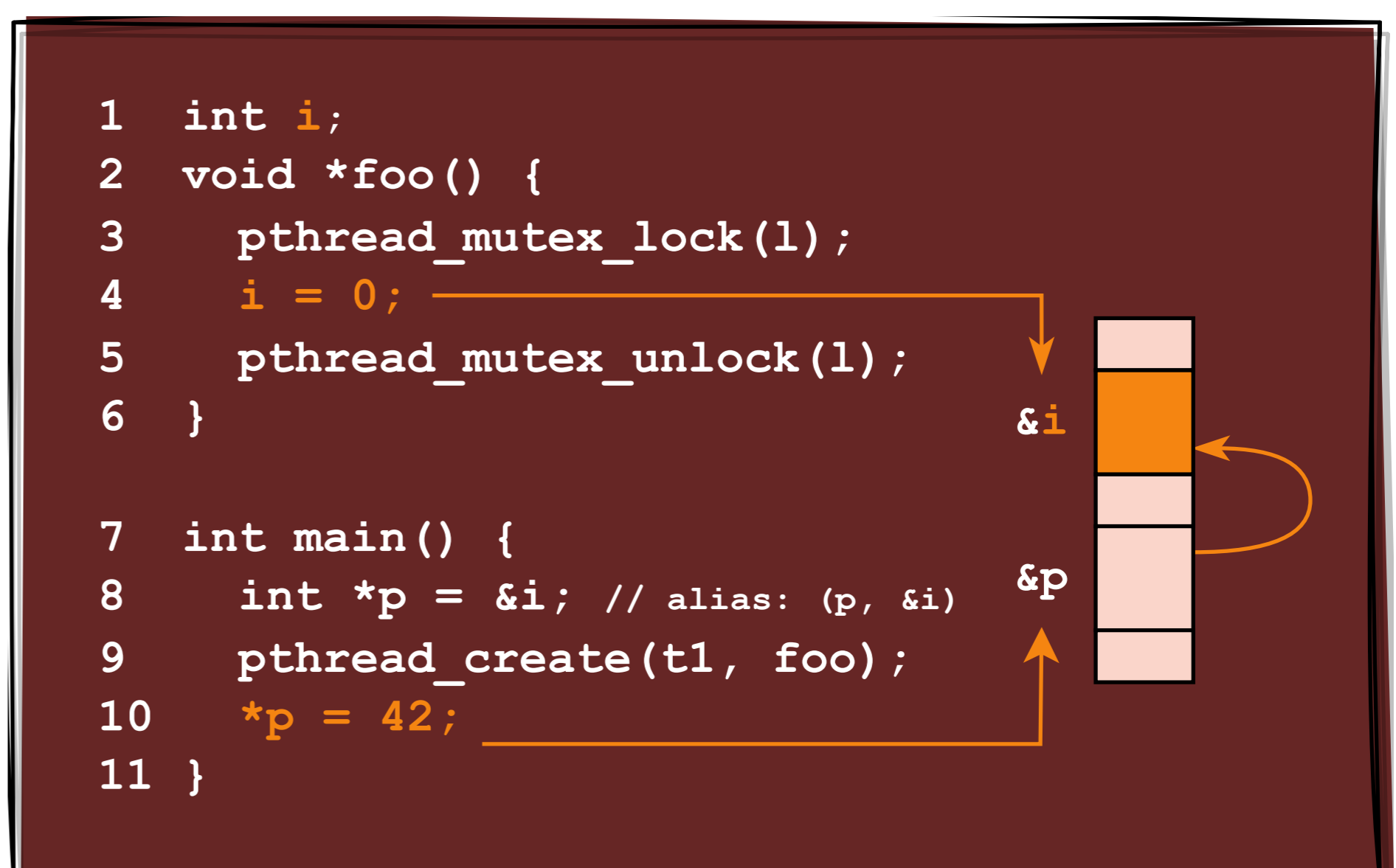
INFER: Open Source Static Analysis Framework from Meta/Facebook



- Open-source tool for static analysis
- Based on abstract interpretation
- Highly scalable, compositional, and incremental
- Analyses code changes only
- Checkers for various kinds of bugs
 - null-dereferencing, memory leaks, buffer overflow...
 - data race detection for Java only
- Easy to extend with new analyser plugins
- Supports C, C++, Objective-C, and Java

DarC: A New Data Race Checker for INFER

- Designed for programs written in C/Pthreads
- Inspired by the **RacerX** [1] and **RacerD** [2] analysers
- Computes function **summaries**:
summary: (accesses, threadset, lockset, aliases)
- Detects data races between pairs of **accesses**:
access: (var, loc, type, threadset, lockset, thread)
- A data race is reported if all of the following conditions hold for a pair of accesses (a, a'):
 1. a.var = a'.var
 2. a.type = Write \vee a'.type = Write
 3. a.thread \neq a'.thread
 4. a.lockset \cap a'.lockset = \emptyset
 5. a.thread \in (a.threadset \cap a'.threadset) \vee a'.thread \in (a.threadset \cap a'.threadset)



accesses: {(i,4,Write,{main,t1},{1},t1),
(i,10,Write,{main,t1},{},main)}
→ DATA RACE

Experimental Evaluation

- Experiments on both simple and more complex programs written in C/Pthreads
- Comparison with the static analyser **CODERRECT** and the dynamic analysers **THREADSANITIZER** and **HELGRIND**
- **DataRaceBenchmark** [3]:
 - benchmark for evaluating concurrency testing tools
 - consists of 67 programs (38 – 257 LOC)
- **ConcurrencyBenchmark** [4]:
 - small tests developed for testing DarC
 - consists of 82 programs (15 – 88 LOC)

analyser	DataRaceBenchmark				ConcurrencyBenchmark			
	races	no races	errors	time	races	no races	errors	time
DARC	41	26	0	26.3s	55	27	0	30s
CODERRECT	29	33	1	1m20s	24	55	3	1m40s
THREADSANITIZER	40	23	0	17.7s	45	36	0	19.53s
HELGRIND	40	22	1	28.5s	41	39	1	36.24s

References

- [1] Dawson Engler and Ken Ashcraft. Racerx: Effective, static detection of race conditions and deadlocks. In *Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles, SOSP '03*, page 237–252, New York, NY, USA, 2003. Association for Computing Machinery.
- [2] S. Blackshear, N. Gorogiannis, P. O'Hearn, and I. Sergey. RacerD: Compositional Static Race Detection. *Proceedings of the ACM on Programming Languages*, 2(OOPSLA):144:1–144:28, 2018.
- [3] DataRaceBenchmark. <https://github.com/marchartung/DataRaceBenchmark>.
- [4] ConcurrencyBenchmark. <https://github.com/svobodovaLucie/ConcurrencyBenchmark>.