# Searching for and fixing UB across all of crates.io

github.com/saethlin/miri-tools

miri.saethlin.dev/ub

# UB is the essence of unsafe programming

Optimizations assume that unsafe preconditions are met

Basically all programs are built on unsafe foundations

UB is very often a time bomb

# I'm not that smart, that's my primary benefit

Miri

Standard library debug assertions, -Zbuild-std

Address Sanitizer

I just run people's tests

# Miri findings

4,836 of ~86,000 crates report UB under Miri

~1,600 are some kind of pointer aliasing problem

~1,000 are mem::uninitialized or invalid MaybeUninit::assume_init

~720 are https://github.com/rust-lang/rust-bindgen/issues/1651

~150 are misaligned pointer access

# Miri findings (cont)

35 use after free

14 data race

1 or 2 of so many other things...

# Standard Library Debug Assertions findings

~10,000 crates

18 out-of-bounds get_unchecked

11 misaligned pointer

10 null slice

3 copy_nonoverlapping called with overlapping ranges

# AddressSanitizer findings

~10,000 crates

5 Buffer overflow in manual SIMD code

3 Null terminator confusion

3 Insta-dangling pointer

1 Use after free

# Fixing UB

So far I have sent PRs to ~50 crates

Most maintainers take UB-fixing patches quickly

Crates rarely contain just one bug

Sending PRs is way more work than finding problems

# Advice?

Rust-specific UB dominates

```
RUSTFLAGS=-Zsanitizer=address cargo test -Zbuild-std

cargo miri test
```