PRTest: A plain random tester

Thomas Lemberger

LMU Munich, Germany
Evaluation of new testing techniques:
  ▶ Comparison to pre-decessors
  ▶ Comparison to tools with different strengths/weaknesses
  ▶ No comparison to naive/worst-case approach

⇒ Create such a naive tool for Test-Comp!
Instrument program with custom input-method
Execute instrumented program indefinitely
Monitor branch coverage to decide which tests to keep
Weaknesses of PRTEST

```java
if (input() == 1) {
    // interesting code
}
```

- Probability to create fitting test: \( \frac{1}{2^{32}} = 0.000000002\% \)
- Speed of PRTEST strongly depends on execution speed of program under test
Strengths of PR\textsc{test}

▶ Very fast to create initial test suite
▶ Supports all language constructs and features of C
▶ Very simple $\Rightarrow$ good baseline
Categories in which PRTEST was not last place:

► Coverage-Error:
  ► Arrays (7th/9)
  ► Floats (3rd/9)
  ► Heap (7th/9)

► Coverage-Branches:
  ► Floats (4th/7)
Repository

▶ https://github.com/sosy-lab/tbf/tree/master/tbf/tools/random