

M10- Equivalence Modulo Input

Guillermo Polito

ECI'23 - Universidad de Buenos Aires

Goals

- Exploit metamorphic relations to derive oracles
- Using real world programs to test compilers at scale
- Semantic-preserving mutations as sources of metamorphic relations for compilers

Remember Date Fuzzer

```
(1 to: 10) collect: [ :e | (PzGrammarFuzzer on: PzDateGrammar new) fuzz ]
```

23 5
7/February-6
7,February0
0/february/7
9 february 0
7 February-9
February 0,1
4/February,4
february/0 7
1January,8

**How can we produce an oracle
for dates ?**

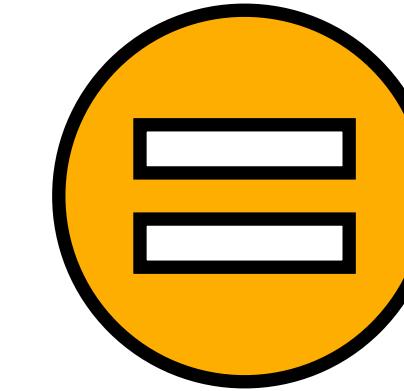
Same Date has Different Representations

10-2-03

10-02-2003

Differential Testing on Different Inputs

'10-2-03' asDate



'10-02-2003' asDate

Metamorphic Relations

$\text{program}(\text{input}) = \text{program}(\text{input}')$
where input' is derived from input

- E.g.,
 - $\sin(x) = \sin(\pi - x)$
 - $\text{factorial}(x) = \text{factorial}(x + 0)$

Metamorphic Relations

$\text{program}(\text{input}) = \text{program}(\text{input}')$
where input' is derived from input

- E.g.,
 - $\sin(x) = \sin(\pi - x)$
 - $\text{factorial}(x) = \text{factorial}(x + 0) \longrightarrow$ Ok, this is not a nice one.
How can we obtain interesting ones?

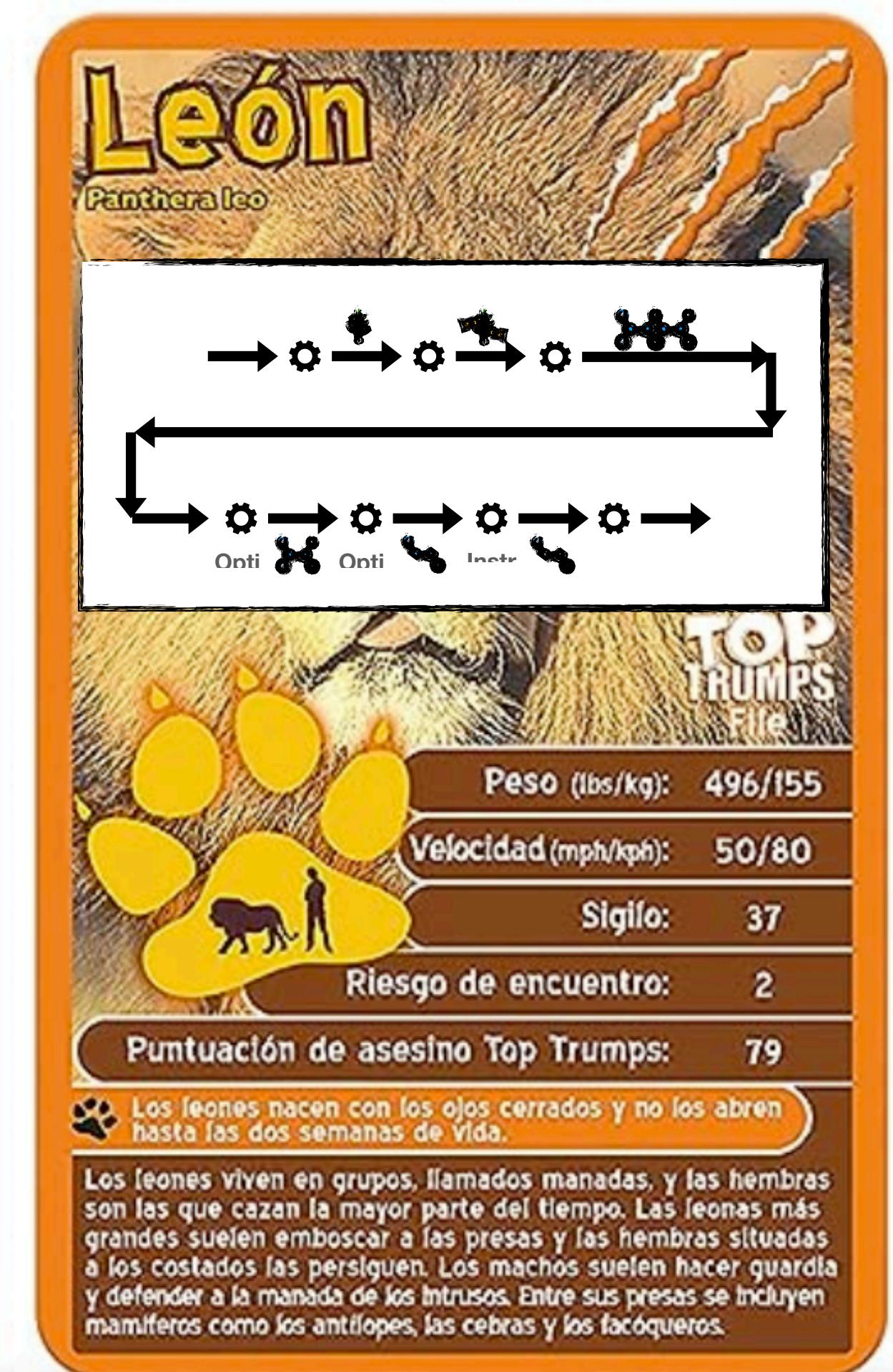
Remember Cross-Optimization Testing

- Same compiler
- + variations
 - e.g., optimizations
 - e.g., architecture
- On the same program

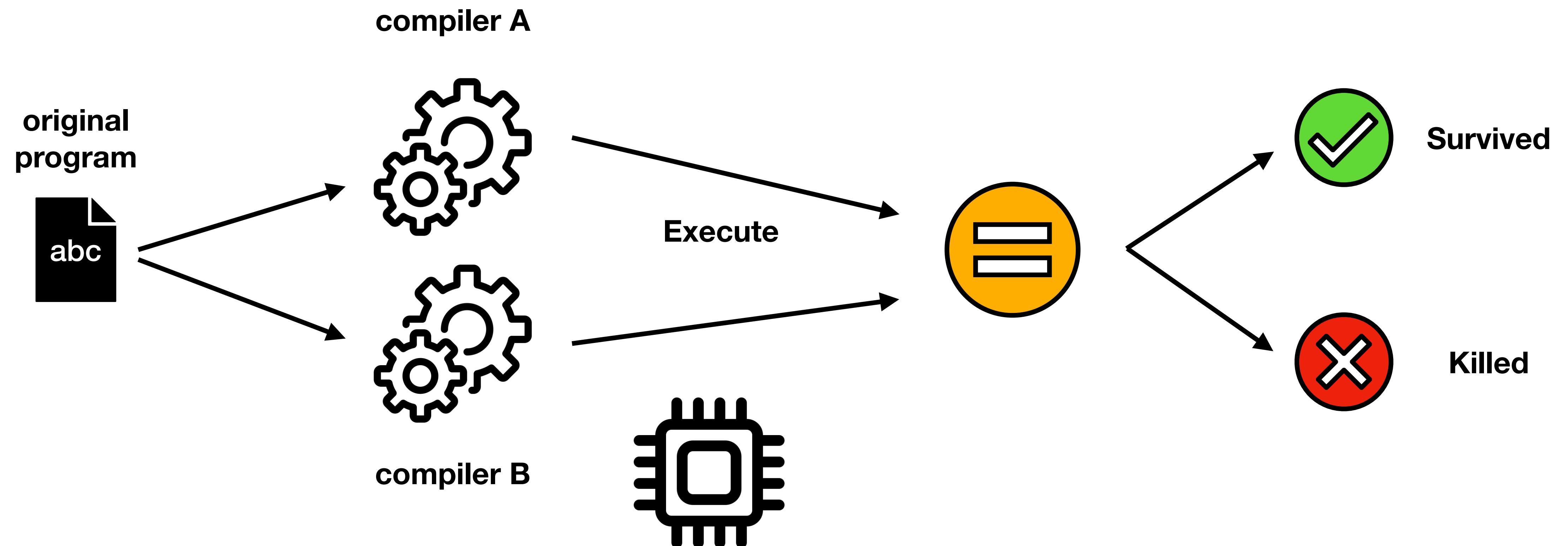
O2 level



O0 level

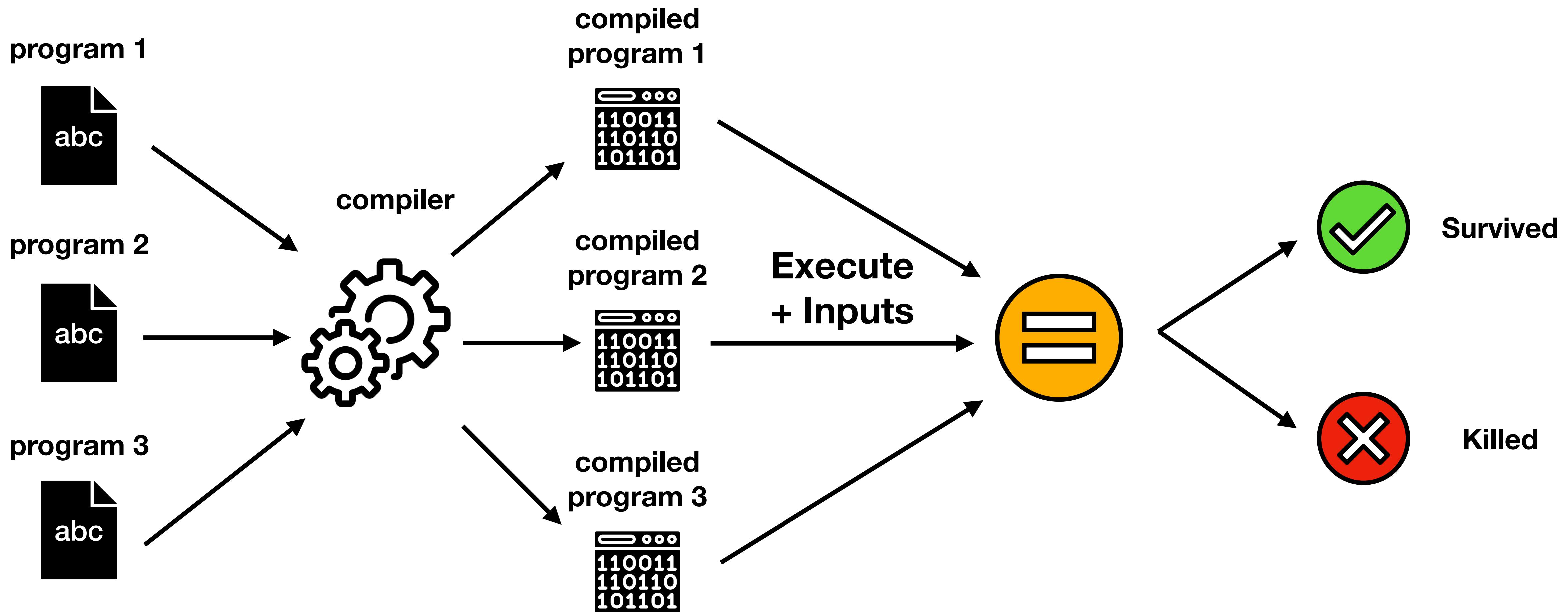


Differential Testing on Compiler Variations



How can we test with the same compiler (*variation*) ?

Differential Testing on Program Variations



How can we generate
semantically equivalent
programs?

How can we generate semantically equivalent programs?

Or, what metamorphic relations
do compilers have?

Semantic-preserving Mutations

- Remove dead code
- Equivalent expressions
 - $e * 2 \Rightarrow e + e$
 - $(a + b) * 3 \Rightarrow a * 3 + b * 3$
- Reordering statements – but keeping semantics
- Pre-optimized code

Takeaways

- Differential testing can be applied to inputs too
 - Metamorphic relations allow deriving oracles from inputs
 - Extracting metamorphic relations is challenging
-
- For compilers: Semantic-preserving mutations

Material

- An Automatic Testing Approach for Compiler Based on Metamorphic Testing Technique. APSEC'10
Qiuming Tao et al.
- Metamorphic Testing: A Review of Challenges and Opportunities. ACM Computing Surveys'18.
Tsong Yeah Chen et al.
- Compiler Validation via Equivalence Modulo Inputs. PLDI'14
Vu Le et al.