

Compiler Testing

Software Evolution – L4P1

Dr. Vadim Zaytsev aka @grammarware, February/March 2021



Compiler Testing Books

Testing the Front-End [1/3]

- **R**-tests and **F**-tests
 - known input
 - expect the parser to terminate
- BabyCobol
 - positioning
 - line continuations
 - whitespace
 - . . .

Testing the Front-End [2/3]

- **P**-tests
 - known input
 - known output tree structure
- BabyCobol
 - expressions (also contracted)
 - sentences
 - feature interplay corner cases
 - . . .

Testing the Front-End [3/3]

- **Q**-tests
 - some input
 - *unparse*(*parse*(text)) = text
 - *parse*(*unparse*(tree)) = tree
- BabyCobol
 - like for **R** or **P** but less work
 - . . .

Testing the Middle “End” [1/2]

- **N**-tests
 - some input with a known fragment
 - canonical form of that fragment
- BabyCobol
 - contracted expressions
 - sufficient qualification
 - . . .

Testing the Middle “End” [2/2]

- **T**-tests and **E**-tests
 - some input with typed content
 - type system conformance
- BabyCobol
 - invalid MOVE, EVALUATE, SUBTRACT, ...
 - DATA DIVISION invalidating MOVES
 - implicit types of undeclared fields
 - . . .

Testing the Back-End [1/4]

- **C**-tests and **U**-tests
 - known input
 - compilability
- BabyCobol
 - consistent mini-programs

Testing the Back-End [2/4]

- **V**-tests
 - known input
 - validity of generated or interpreted code
- BabyCobol
 - input errors do not lead to crashes

Testing the Back-End [3/4]

- **S**-tests and **X**-tests
 - known input
 - executability of compiled code
- BabyCobol
 - consistent mini-programs

Testing the Back-End [4/4]

- **D**-tests
 - direct call to a runtime function
 - known expected output
- BabyCobol
 - anything that ends up in runtime
- (This is normal testing)

Conclusion

- Front-end: R, F, P, Q
- Middle "end": N, T, E
- Back-end: C, U, V, S, X, D
- Q&A Sessions @ Canvas
 - ⇒ v.zaytsev@utwente.nl
 - ⇒ <https://discord.gg/n7VQAPNBPD>