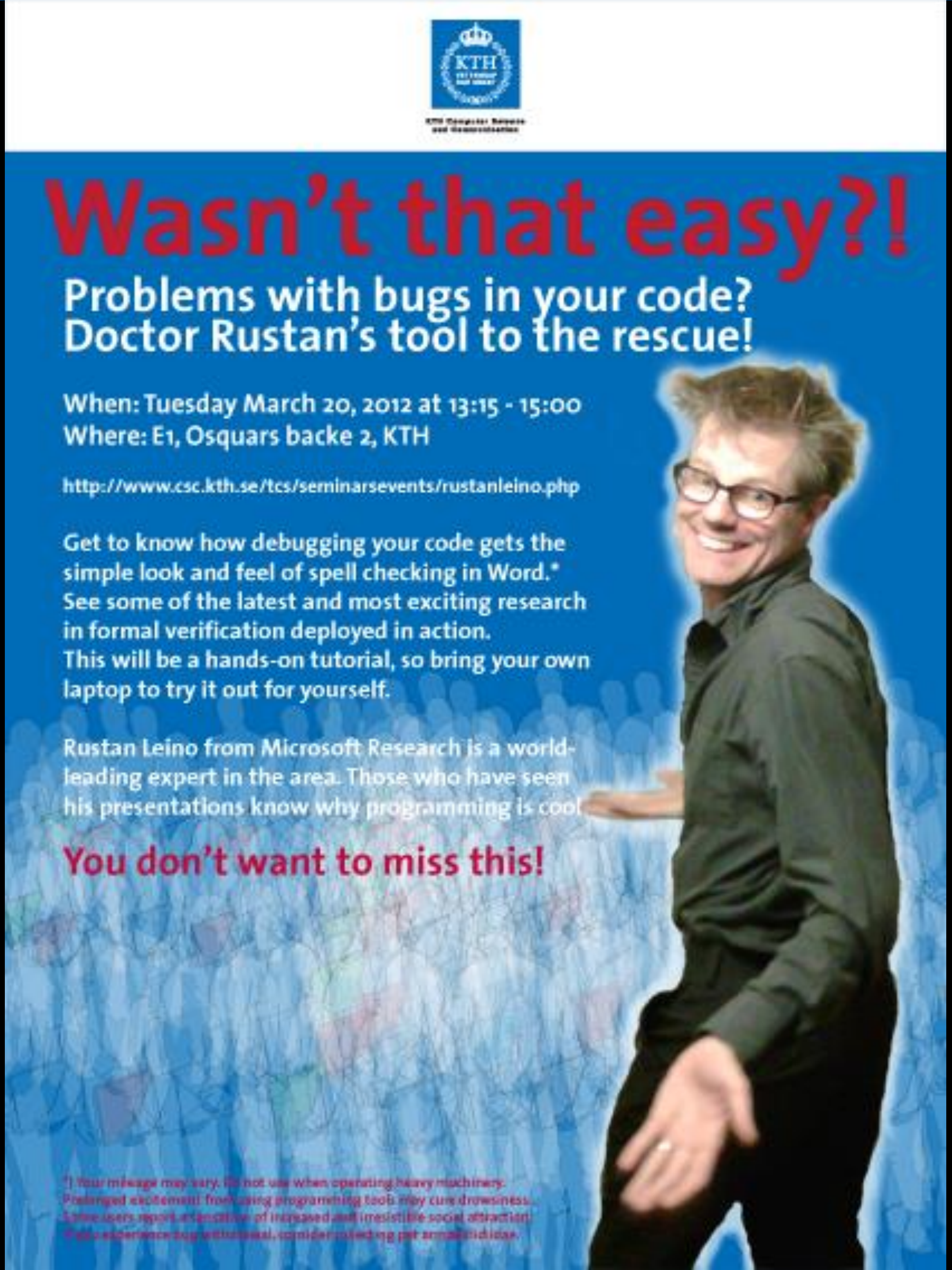
Dafny

Testing, Quality Assurance, and Maintanance
Fall 2023
Prof. Arie Gurfinkel

based on slides by K. Rustan M. Leino and Gudmund Grov

UNIVERSITY OF WATERLOO



KTH Computer Science and Communication

Wasn't that easy?!

Problems with bugs in your code? Doctor Rustan's tool to the rescue!

When: Tuesday March 20, 2012 at 13:15 - 15:00
Where: E1, Osquars backe 2, KTH

http://www.csc.kth.se/tcs/seminarsevents/rustanleino.php

Get to know how debugging your code gets the simple look and feel of spell checking in Word.* See some of the latest and most exciting research in formal verification deployed in action. This will be a hands-on tutorial, so bring your own laptop to try it out for yourself.

Rustan Leino from Microsoft Research is a world-leading expert in the area. Those who have seen his presentations know why programming is cool

You don't want to miss this!

*) Your mileage may vary. Do not use when operating heavy machinery. Prolonged excitement from using programming tools may cure drowsiness. Some users report a sensation of increased and irresistible social attraction. If you experience bug withdrawal, consider collecting pet armadillidiae.

Dafny

Programming language designed for *reasoning*

Language features drawn from:

- Imperative programming
  - `if`, `while`, :=, `class`, …
- Functional programming
  - `function`, `datatype`, `codatatype`, …
- Proof authoring
  - `lemma`, `calc`, `refines`, `inductive predicate`, …

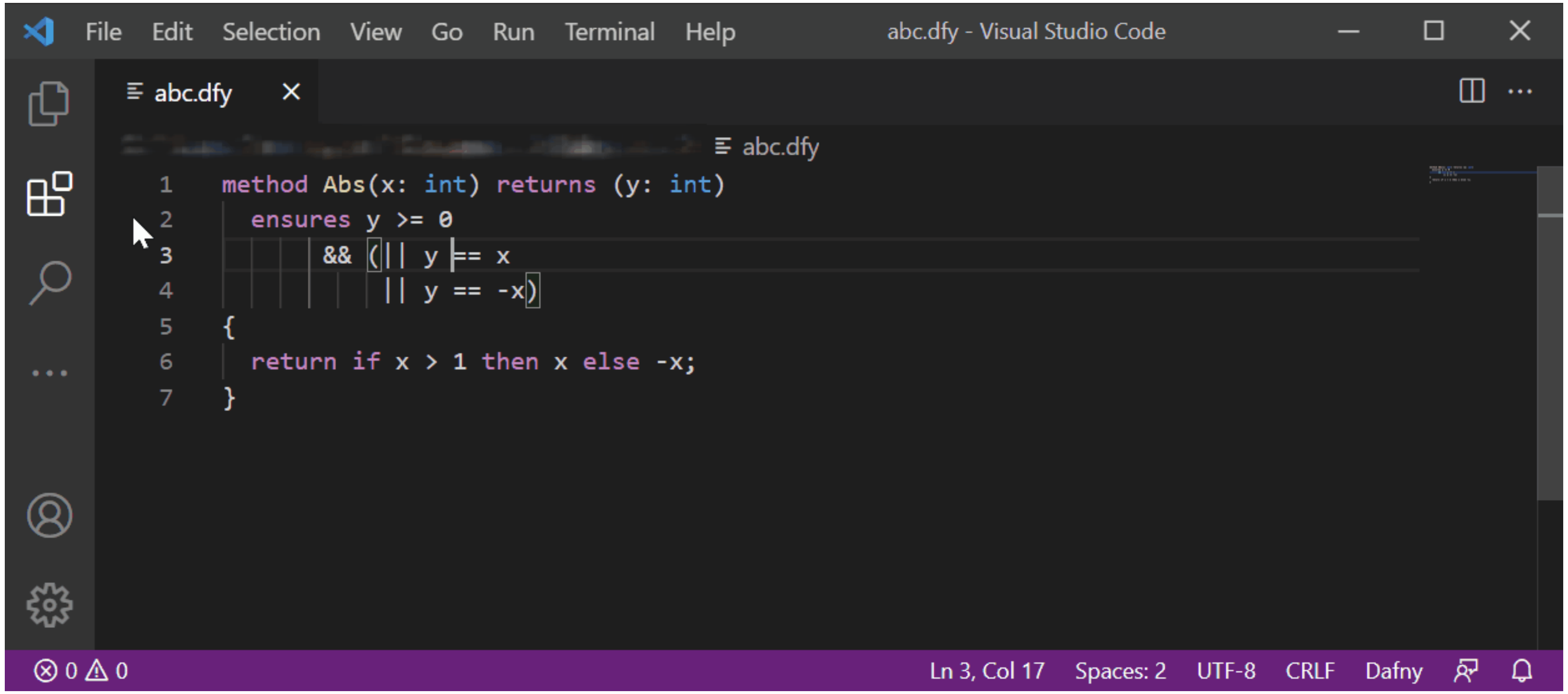
Program verifier

Integrated development environment (IDE)

Using Dafny

Dafny plugin in VSCode

(Dafny mode in Emacs)

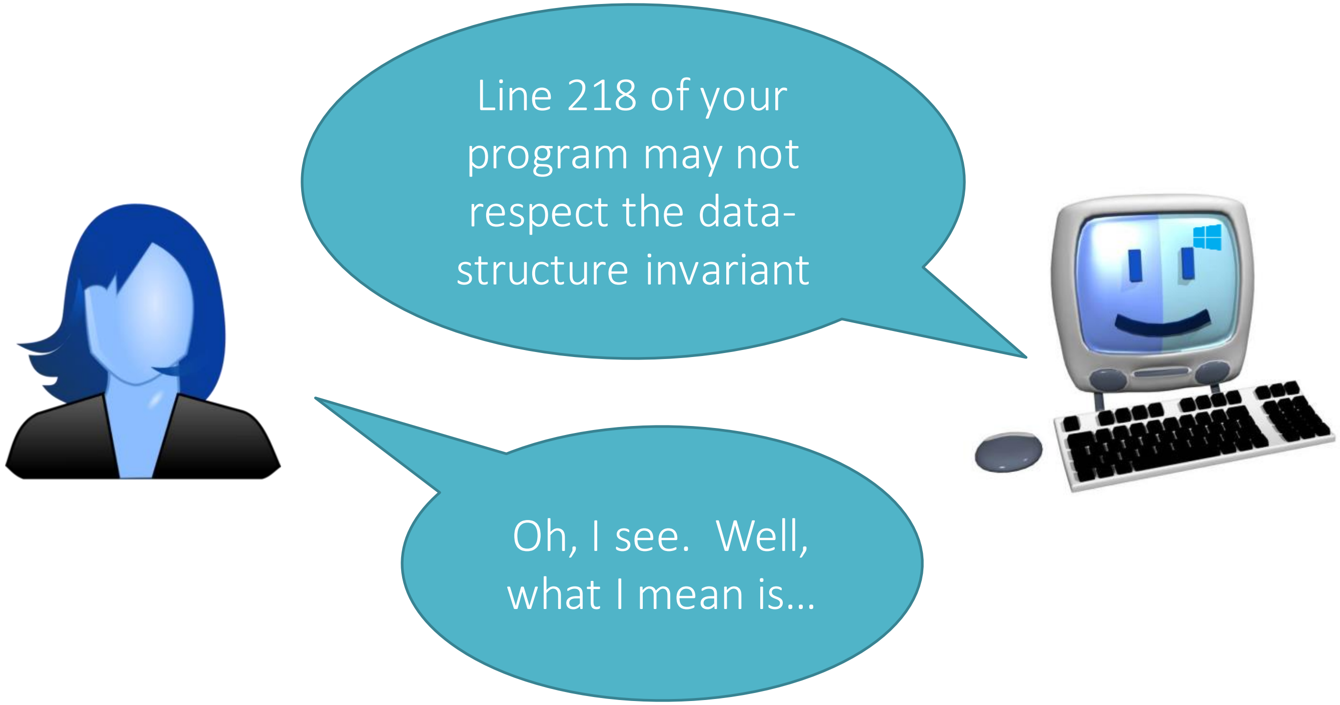


```
method Abs(x: int) returns (y: int)
  ensures y >= 0
       && (|| y == x
           || y == -x)
{
  return if x > 1 then x else -x;
}
```

https://github.com/dafny-lang/dafny

Involving the programmer

- Opportunities
  - Tool’s analysis can be customized and targeted
  - Allows interaction with tool, like a programmer’s apprentice


Line 218 of your program may not respect the data-structure invariant
Oh, I see. Well, what I mean is…

Projects that involve the programmer

Paris Metro line 14 brake system (B)

seL4 Verified (Haskell, Isabelle/HOL, C)

CompCert (Coq)

Ironclad (Dafny)

…

Common among these projects:

- Tool is part of development process
- Specifications, code, proofs developed together
- No legacy code

Involving the programmer

- Paris Metro line 14 brake system (B)
- seL4 Verified (Haskell, Isabelle/HOL, C)
- CompCert (Coq)

Verification done by formal-method experts

- Ironclad (Dafny)

Verification done by systems programmers

Uses of Dafny

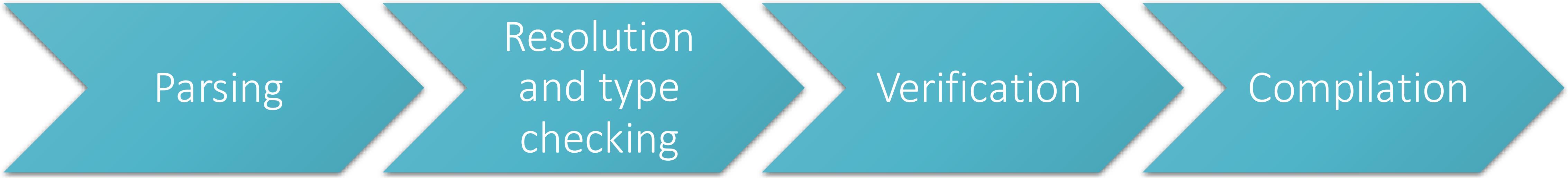
In projects

- ExpressOS [ASPLOS 2013]
- CloudMake algorithms [FM 2014]
- Ironclad Apps [OSDI 2014]
- IronFleet [SOSP 2015]

In teaching

- At over 30 universities

Dafny pipeline



Parsing

Resolution and type checking

Verification

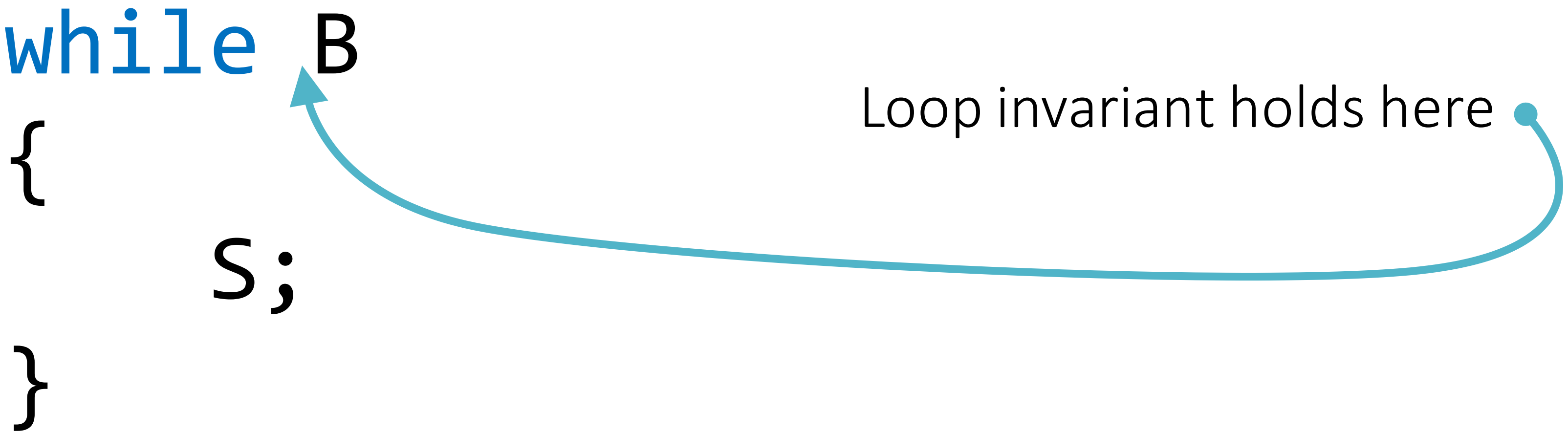
Compilation

Reasoning about loops

A loop invariant

- holds at the top of every iteration
- is the *only* thing the verifier remembers from one iteration to another (about the variables being modified)

It is as if the loop body were not available



```
while B
{
      S;
}
```

Loop invariant holds here

Conclusions

Functional-correctness verification is becoming more automatic

Dafny

- Use
- Teach
- Extend

research.microsoft.com/dafny

- Papers

github.com/dafny-lang/dafny

- Binaries
- Sources
- Discussion forum

research.microsoft.com/verificationcorner

- Videos -- now on YouTube