

*Megan Fowler*

<https://doi.org/10.1145/3328778.3372715>

## **ABSTRACT**

One aspect of developing correct code, code that functions as specified, is annotating loops with suitable invariants. Loop invariants are useful for human reasoning and are necessary for tool-assisted automated reasoning. Writing loop invariants can be a difficult task for all students, especially beginning software engineering students. In helping students learn to write adequate invariants, we need to understand not only what errors they make, but also why they make them. This poster discusses the use of a Web IDE backed by the RESOLVE verification engine to aid students in developing loop invariants and to collect performance data. In addition to collecting submitted invariant answers, students are asked to provide their steps or thought processes regarding how they arrived at their answers for each submission. The answers and reasons are then analyzed using a mixed-methods approach. Resulting categories of answers indicate that students are able to use formal method concepts with which they are already familiar, such as, pre and post-conditions as a starting place to develop adequate loop invariants. Additionally, some common trouble spots in learning to write invariants are identified. The results will be useful to guide classroom instruction and automated tutoring.