

**ZMATH 2016f.01353**

**Drachova, Svetlana V.; Hallstrom, Jason O.; Hollingsworth, Joseph E.; Krone, Joan; Pak, Rich; Sitaraman, Murali**

**Teaching mathematical reasoning principles for software correctness and its assessment.**

ACM Trans. Comput. Educ. 15, No. 3, Article No. 15, 22 p., electronic only (2015).

Summary: Undergraduate computer science students need to learn analytical reasoning skills to develop high-quality software and to understand why the software they develop works as specified. To accomplish this central educational objective, this article describes a systematic process of introducing reasoning skills into the curriculum and assessing how well students have learned those skills. To facilitate assessment, a comprehensive inventory of principles for reasoning about correctness that captures the finer details of basic skills that students need to learn has been defined and used. The principles can be taught at various levels of depth across the curriculum in a variety of courses. The use of a particular instructional process is illustrated to inculcate reasoning principles across several iterations of a sophomore-level development foundations course and a junior-level software engineering course. The article summarizes how learning outcomes motivated by the inventory of reasoning principles lead to questions that in turn form the basis for a careful analysis of student understanding and for fine-tuning teaching interventions that together facilitate continuous improvements to instruction.

*Classification:* P55

*Keywords:* evaluation; components; formal methods; learning outcomes; objects; specifications

doi:10.1145/2716316