DOI 10.1007/s00165-009-0109-8

Formal Aspects of Computing (2009) 21: 225

Formal Aspects of Computing

Editorial

This special issue on education has its origins in two workshops, FM-Ed 2006[1] (Formal Methods in the Teaching Lab) held in Hamilton, Canada, in August 2006 and Teaching Formal Methods 2006[2] (TFM 2006) held in London, UK, in December 2006. The former was organised by the Formal Methods Europe Subgroup on Education and the latter by the British Computer Society Formal Aspects of Computing Science (FACS) specialist group and Oxford Brookes University. The proceedings of both workshops were published on-line. After the workshops, authors were invited to submit significantly extended papers for inclusion in this special issue. Following thorough review, five of the submissions were accepted.

The paper by Back, "Invariant based programming: basic approach and teaching experiences" introduces a diagrammatic approach to teaching invariant based programming—an approach that aims to produce programs and software that are correct by construction. The author describes the approach, then contrasts the experience of teaching invariant based programming to experienced programmers and to novice programmers.

Larson, Fitzgerald and Riddle, "Practice-oriented Courses in Formal Methods using VDM^{++}" present a practical approach to using a formal method to aid computing science students in developing skills in abstraction and rigorous analysis. Examples drawn from industrial projects provide motivation to students and the courses are taught using industrial-strength tool sets.

The remaining three papers deal directly with tools. Tsay et al., "Tool Support for Learning Büchi Automata and Linear Temporal Logic", describes GOAL (Graphical Tool for Omega-Automata and Logics) an interactive graphical tool designed to help students to understand Büchi automata, linear logics and the relationship between them. Functionality includes translation from a temporal formula to an automaton and visualization of how the (non-deterministic) automaton runs on a given input.

"The RISC ProofNavigator: A Proving Assistant for Program Verification in the Classroom" by Schreiner, describes the proof tool and its graphical user interface—designed to make the tool suitable for educational use—together with evaluation through a number of use cases.

The final paper, "A Comparison of Tools for Teaching Formal Software Verification", by Feinerer and Salzer, describes how four tools for formal verification were evaluated for use in class and reports experience of applying the chosen tool in this context.

As organisers of the two workshops, we would like to thank 'Formal Aspects of Computing' for allowing us to publish this collection of papers in the journal, and also the Editor-in-Chief and Associate Editor for their support and assistance as the collection was assembled. We also wish to thank the participants in the two workshops for providing the stimulating context in which the original papers were presented, and the individuals who reviewed the submissions to this special issue.

Paul Boca
Raymond Boute
David Duce
José Oliveira

Published online 26 March 2009

Correspondence and offprint requests to: D. Duce, E-mail: daduce@brookes.ac.uk

[1] FM-Ed 2006, `http://www.di.uminho.pt/FME-SoE/FMEd06/`

[2] Teaching Formal Methods 2006, `http://www.bcs.org/ewic/tfm2006/`