Abstract

Programming is a practical process; students need to write a lot of programs in order to master it. However, with large number of students, the assessment of programming exercises leads to extensive workload for teachers making it difficult for instructors to provide constructive and corrective feedback or even additional help when the students need it. In this work, we address the issue of automatic assessment for programming assignments. The goal of which is to provide immediate grading and comprehensible feedback to the learners, while taking some of the workload burden off the teachers. This paper proposes a system combining results from dynamic and static analysis to ensure a reliable and objective evaluation job. While dynamic analysis is based on unit testing framework, the static analysis will quantify the structural similarity between students’ programs and the solutions provided by the teacher. In order to perform such comparison, a suitable program representation and an adequate similarity measure will be presented.
References

1. Scherer, R., Siddiq, F., & Sánchez Viveros, B, The cognitive benefits of learning computer programming: A meta-analysis of transfer effects, Journal of Educational Psychology. Advance online publication, 2018,
2. K. Heggar, Coded for success: the benefits of programming among school students, June 2014,
3. Higgins, S., Hall, E., Baumfield, V., Moseley, A meta-analysis of the impact of the implementation of thinking skills approaches on pupils, Research Evidence in Education Library, 2005,
5. J. Hollingsworth. Automatic graders for programming classes, Communications of the ACM, 3:528–529, October 1960,
9. Soundous Zougari, Mariam Tanana, Abdelouahid Lyhyaoui, Towards an automatic assessment system in introductory programming courses, 2016,
13. Mohammad Shaikat Amin, Russell L. Finley, Hasan M. Jamil, Top-k Similar Graph Matching Using TraM in Biological Networks, 2012,
15. Vincenzo CARLETTI, Exact and Inexact Methods for Graph Similarity in Structural Pattern Recognition, 2016,
16. Danai Koutra et al., Algorithms for Graph Similarity and Subgraph Matching, 2011,
18. MilenaVujošević-Janičića, Mladen Nikolić, Dušan Tošić, Viktor Kuncakb, Software verification and graph similarity for automated evaluation of students’ assignments, Information and Software Technology Vol 55, 2013,
Index Terms

Computer Science                      Information Sciences

Keywords

Programming assessment, dynamic analysis, static analysis, CFG similarity measure, automated grading.