ABSTRACT

This work develops a new method of program assessment based on equivalence checking of finite state machines with data-paths (FSMDs), reporting the error and providing feedback towards error correction. In order to assess a student's program, it is compared with a model program supplied by the teacher or instructor. For comparison, we actually establish the equivalence of their FSMDs, which is an intermediate representation of the programs and as FSMD is a graph like a flowchart, where the vertices are states and edges represent data and control flow, it provides scope for further analysis of paths in terms of statement containment, which is done after the execution of equivalence checker reports error. It uses containment checking algorithm, by which it categorizes the containment in the FSMD of a student's program into one of the three types of cases identified in this work, in order to assess the student's program. These categories of containment can be mapped to various types of errors in the programs with respect to the golden model, which may creep in while programming. The details of these errors and the methods for detecting and reporting them are described in this thesis. Various cases of errors in the students' programs have been correctly analyzed.

Conditional constructs require the conditions to follow a precedence order, so that the code corresponding to no condition is un-reachable. If there is a violation of precedence in occurrence of conditions, then the program can be immediately declared to be erroneous. This work describes how handling conditional constructs could be done as a pre-processing step to the equivalence checking to facilitate better feedback to the student. This work further addresses the variable mapping problem, as the reference program is likely to use variable names that are quite different to those used in the students' programs. This problem is important as the equivalence checking method requires that the names of the variables used must be the same, in the programs whose equivalence is to be established.

Next in the thesis, the automated checking of approximate equivalence of expressions, that cannot effectively be checked through formal equivalence checking, is described. It is based on a randomised simulation in a given domain. The final topic covered is an automated evaluation scheme for awarding marks to student programs.

Keywords: *FSMD, equivalence checker, containment checking, cut-point, last correct state (LCS), corresponding state of last correct state (CSLCS).*