



Software Engineering Seminar

Test Case Generation/Selection

Description

Recognition of a bug is the first step to be able to fix it. As, generally, there does not exist a formal specification for the software system under consideration, a *test suite* is being used to help ensure the correctness of the system. When executing a test suite, a failing test case is an indication of an error that resides in the system.

A problem associated with test suites is that someone has to actually create, apply and maintain them. To this end, *automated techniques* for test case generation and selection have been developed in order to achieve various test goals (e.g., maximum line/branch coverage) which can be specified by the user. The goal of this topic is to obtain an overview over existing techniques for automated test case generation and selection and to discuss and evaluate them.

References

- [1] Shay Artzi, Julian Dolby, Frank Tip, and Marco Pistoia. Directed test generation for effective fault localization. *Proceedings of the 2010 international symposium on Software testing and analysis - ISSA '10*, pages 49–59, 2010.
- [2] Sebastian Elbaum, Alexey G. Malishevsky, and Gregg Rothermel. Prioritizing Test Cases For Regression Testing. *International Symposium on Software Testing and Analysis*, pages 102–112, 2000.
- [3] S. Rayadurgam and M.P.E. Heimdahl. Coverage based test-case generation using model checkers. *Proceedings. Eighth Annual IEEE International Conference and Workshop On the Engineering of Computer Based Systems-ECBS 2001*, pages 83–91, 2001.
- [4] Gregg Rothermel and Mary Jean Harrold. A Safe , Efficient Regression Test Selection Technique. *ACM Transactions on Software Engineering and Methodology*, 6(2):173–210, 1997.

Contacts

Simon Heiden (heiden@informatik.hu-berlin.de)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin