Testing and Maintenance in Software Engineering Education

Discussion Summary

Geoffrey Kennedy
Information Science Department, University of Otago, Dunedin, New Zealand

1 Objectives of the Group

The testing of software appears to be based on heuristics arrived at by individual developers over a lifetime of experience. This was the consensus of the group who came together to discuss this issue. There appears to be no properly defined discipline or standard set of approaches which educationalists can teach to students of software engineering. Having come to this rather daunting conclusion some attempt was made to at least identify whatever wisdom the group members had to offer.

2 Experiences from within the Group

When asked what they were expecting to take away from the discussion, the common reply was to find a way of making students aware of the problem, aware of the need for careful and systematic testing. All participants had experienced the frustration of encountering students who felt that they were well able to test their products, but who failed often to apply the most basic of testing strategies.

The question of software maintenance was also raised, but reluctantly set aside with the realisation that, within the time constraints of the typical Polytechnic and University courses, meaningful longitudinal studies which might highlight some of the problems to be overcome. Some participants had used student work from previous years as a source, but found a lack of purpose and direction for subsequent students to pursue.

3 Towards a more systematic approach to testing

The important link between quality assurance and testing strategies was noted and the possibility of including testing considerations as an integral part of the system requirements specification was explored. Apart from the fact that it would add to the complexity of an already difficult task it was generally applauded. It was further noted that in many cases the exact testing strategies might depend on the type of system being developed, for example time or performance critical, or upon the development tool in use, as for example fourth generation productivity tools.

4 Conclusions

Clearly the problems of teaching about program and system testing in Software Engineering courses were deeply felt by the participants. The revelation that there appears to be a lack of a clearly defined discipline certainly helps us understand the problem; hopefully it should promote some feverish research activity in this direction. Certainly if we had a clear cut discipline it would help us to teach it. But then, it might be that the problem lies in the nature of the testing activity itself. In this case we will be forced back to our heuristics and favourite test cases.

Participants:
Robert Cohen, University of Newcastle, NSW
Brian Egerton, Nelson Polytechnic, NZ
Dan Hawthorn, Unitec, Auckland, NZ
Leo Holmes, Central Inst. of Tech., NZ
Geoffrey Kennedy, University of Otago, NZ
Maryam Purvis, University of Otago, NZ