Propositions

pertaining to the thesis

Reasoning about object-oriented programs: from classes to interfaces

by Jinting Bian

1. The main bottleneck in the verification process is not the verification itself, but the formulation of specifications. [Chapter 3]

2. The history-based reasoning approach provides a way to show the satisfiability of specifications by a witness implementation of the interface, making it possible to reason about the state-hidden interface. [Chapter 4]

3. The selection of abstractions of history in a concrete program requires careful consideration. [Chapter 5 & 6]

4. It is necessary to verify the correctness of the subtype relation in the design stage, especially in the case of complex systems. [Chapter 7]

5. Although the cost upfront for ensuring program correctness is expensive and the benefits come late (even after time to market), it is still worthy.

6. The correctness of the theorem prover used in a formal verification requires careful attention.

7. Testing can reveal the presence of faults in software, while formal verification aims to prove the absence of failures; both of them are indispensable and irreplaceable.

8. The rapid pace of technological change and the demand for new features pose a significant challenge to the adaptability and effectiveness of formal verification in system development.

9. Intelligent dialogue systems can help in the development of formal methods, yet the techniques and expertise required for formal methods cannot be replaced by artificial intelligence.

10. Open-source projects benefit from community contributions for faster bug identification and resolution, but this doesn't mean they are bug-free.