

The Verifying Compiler, a Grand Challenge for Computing Research

C.A.R. Hoare

Microsoft Research Ltd., Cambridge, UK

The ideas of program verification date back to Turing and von Neumann, who introduced the concept of an assertion as the specification of an interface between parts of a program. The idea of mechanical theorem proving dates back to Leibniz; it has been explored in practice on modern computers by McCarthy, Milner, and many others since. A proposal for ‘a program verifier’, combining these two technologies, was the subject of a Doctoral dissertation by James C. King, submitted at the Carnegie Institute of Technology in 1969.

Early attempts at automatic program verification were premature. But much progress has been made in the last thirty five years, both in hardware capacity and in the software technologies for verification. I suggest that the renewed challenge of an automatic verifying compiler could provide a focus for interaction, cross-fertilisation, advancement and experimental evaluation of all the technologies of interest in this conference.

Perhaps by concerted international effort, we may be able to meet this challenge, only fifty years after it was proposed by Jim King. We only have fifteen years left to do it.