Speech of acceptance

Fellowship of the Computer History Museum.

Tony Hoare   2007

Thank you for inviting me to address you in absence, to express my delight at election among the Fellows of the Computer History Museum, and my regret that I cannot be with you in person. I can also indulge in a little reminiscence, going back to 1960, when I entered the computing industry as a programmer for a small computer manufacturer, Elliott Brothers of London Ltd. I thought this was already too late in joining the computer revolution, because the first main phase of the expansion of computing was over. My boss was only four years older than me, and my boss's boss was not much older. I would have to wait a long time for promotion. Fortunately I was wrong, and computing has continued to expand in scale and importance throughout my working life.

After a few months experience of machine code programming, my bosses at Elliott Brothers felt I was qualified to design a new programming language for the next computer in their range. Perhaps they hoped to exploit my linguistic education at Oxford University in the classical languages Latin and Greek. In fact, my work on programming languages appealed more to my education in modern philosophy, especially my interest in symbolic logic and the foundation of mathematics. The principles of logic were most clearly embodied in the (now classical) language ALGOL 60, which was our choice for implementation on the new machine.

My experience and discussions with other implementers and researchers strengthened my view that programming language design should be based on the same rigorous axiomatic approach adopted in mathematical logic. I also believed that the habit of concentration and logical reasoning that are necessary for successful computer programming could play the same role in advanced education as the study of Latin and Greek in earlier centuries.

I therefore welcomed the opportunity in 1968 to join the Queens University, Belfast, and in 1977 moved to Oxford University. In both Universities I set up undergraduate and graduate degrees, in which the theory and practice of programming play a central role. For my research, I chose to concentrate on axiomatic approaches to programming and programming language design. I knew that the results of my research

would not be applied in industry for at least thirty years; so I felt that as a topic for long-term research, this would last me all my academic life. This time my prediction has been substantially correct. Even a successful award-winning collaborative development project between University and industry, was never repeated in a second project of the same kind.

So when my expected date of retirement arrived in 1999, I was delighted by Roger Needham's offer of employment in Microsoft's Research Division at its Cambridge Laboratory. This has given me the opportunity to see the ways in which the logical analysis of computer programs has been introduced as an almost universal practice among Microsoft developers. The practice of sprinkling assertions in programs is also quite widespread. They are now used mainly as test oracles, but they are just beginning to feature as contracts, codifying the expectations and obligations of the components of a large program.

My ultimate goal has been that programming languages should be based on logical principles. This has certainly not been achieved. Whether it ever will be, I will have to leave the answer to the judgment of history. As a newly elected fellow of the Museum of the History of Computing, I am surely entitled to do that.