Turing, His Machine and Computability

Jurg Kohlas, MSU #
E-mail: juerg.kohlas@unifr.ch

"Everyone who taps at a keyboard is working on an incarnation of a Turing machine."
Time Magazine, 1999

Alan Turing,
St Vincent,
Scott # 2764,
part of sheetlet

In the last number of Philamath John von Neumann is stated to have said, that if somebody precisely describes what a computer can not do, then he would build a machine which does exactly that. Von Neumann surely knew better. In the year 1936 the young British mathematician Alan Turing published the paper "On Computable Numbers with an Application to the Entscheidungsproblem". In this paper Turing defined what it means to compute mechanically and showed that there are real numbers which can not be computed. Based on this result many other precisely defined problems can be stated which can never be solved by mechanical computing. Von Neumann knew Turing's paper and even met Turing at the Institute of Advanced Study in Princeton, where he participated in the design of the EDVAC the first computer with the architecture named after him. So von Neumann must have known that not every precisely stated mathematical problem can be solved by computer.

Alan Turing devised simple schemes of computations, which today are called a Turing machines. Turing very carefully argued in his paper why his machines really compute everything which reasonably can be called computable. Later several other systems describing computing were proposed which proved to have exactly the same power as Turing machines. Even modern Computers or programming systems have the same power, no more, no less. No computer can solve problems which Turing machines can not. This is known as the *Church-Turing Thesis*:

Turing machines or any equivalent system define what mathematically is meant by an effective or algorithmic procedure or a computation.

It is widely accepted in Computer Science that Turing machines capture the essence of computing at its fundamental level and that no computer in the near or far future is to be expected to solve problems which Turing machines cannot. This is confirmed with the newest model of computing, quantum computers.

Turing showed that the set of all Turing machines can be numbered. Thus the set of all Turing machines is countable. As a consequence only a countable set of numbers can be computed. But Cantor showed that the set of real numbers is uncountable. So there must be many of them which can not be computed. ALL of the numbers one normally encounters in real life ARE computable. This includes all rational numbers, all irrational algebraic numbers like square roots, cube roots, sines, cosines, etc. Even many (transcendental) numbers like pi and e are computable. But there are many real numbers with a precise mathematical definition which can be shown to be uncomputable. These numbers have some importance for Computer Science.

Editor's comment: A more technical - yet understandable – 7 page paper can be obtained from the author in hard or electronic copy.