Stephen C. Kleene was a student of Alonzo Church at Princeton University, where he received his Ph.D. in 1934. In 1935 he went to the University of Wisconsin, to which he remained attached (except for a period in 1941–1946) throughout his distinguished career. A member of the National Academy of Sciences and of the American Academy of Arts and Sciences he received the Steele prize from the AMS in 1983. His book Introduction to Metamathematics *has become a classic. His major area of research interest is recursive functions.*

The Role of Logical Investigations in Mathematics since 1930

STEPHEN C. KLEENE

In 1930 I began work for a Ph.D. in mathematics (received January 1934), specializing in logic and foundations. Being asked now for my impressions about mathematics in the United States during my career, I have two observations.

My first observation is that there has been within the American mathematical community literally an explosion in the amount of the work being done on logic and foundations. When I entered the profession of mathematics, one could almost tally on the fingers of one's two hands the universities and colleges that had in their Mathematics departments a person actively working in that area. This is far from the case now.

Almost all the mathematicians working in logic and foundations in the mid-1930s would have joined the Association for Symbolic Logic by its second year 1937. It had in 1937 just over 200 members (about half from Mathematics departments and half from Philosophy departments, and of course some foreigners). In 1986 it had just over 1600 members (including some computer scientists as well as mathematicians and philosophers).

My second observation is that the work in logic and foundations came in the period we are discussing to impinge significantly on other branches of mathematics. Nowadays, algebraists, topologists and real-variable analysts need to pay attention in connection with some of their enterprises to what the modern logicians are saying.

It has happened before in the history of mathematics that new developments cast important light on preexisting mathematical enterprises. In certain cases, the new developments showed that something mathematicians had been attempting to do for a long time cannot succeed.

Thus, after the Italian mathematicians in the 1500s had obtained formulas solving the general cubic and biquadratic equations in terms of radicals, for two centuries leading mathematicians tried in vain to do the like for the quintic equation. Then, in 1826, Abel demonstrated rigorously the unsolvability of the general quintic equation by radicals.

An example with a greater time span is the work on the problem posed by the Greek geometers around three centuries B.C. to trisect an arbitrary angle using only ruler and compass, which was proved to be impossible in the twentieth century A.D.

The example of a new theory showing the impossibility of success in some earlier endeavors which I shall elaborate is concerned with "algorithms." Mathematicians are familiar with "Euclid's greatest common divisor algorithm" for finding the greatest common divisor of two positive integers a and b; and this is incorporated into an algorithm for answering, for any three integers a, b and c, the question whether the equation $ax + by + c = 0$ has an integral solution for x and y. The term "algorithm" is derived from the name of the ninth century Arabian mathematician al-Khuwarizmi.

An "algorithm" is a certain kind of method for answering any one of a class of questions. The cases we are primarily interested in are ones with the class countably infinite. For example, the Euclidean greatest common divisor algorithm answers the questions "What is the value of $f(a, b)$?" where a and b are any two positive integers and $f(a, b) = \{$the greatest common divisor of a and $b\}$. In the example of solutions of equations $ax + by + c = 0$, the questions are "Does $R(a, b, c)$ hold?" where a, b and c are any three integers and $R(a, b, c) \equiv \{$there exists a solution in integers x and y for the equation $ax + by + c = 0\}$. An "algorithm" for such a class of questions is a method, fully described *before* we pick any particular question from the class, such that the following is the case. *After* we have picked any question (in our two examples, by choosing particular positive integers as values of a, b, or three integers as values of a, b, c), the method will apply, telling us how to perform a succession of discrete steps in some symbolism, after each of which we either have before us the answer to the question and know it, or are told what step to perform next, such that ultimately (after finitely many steps) we will know the answer to the question selected. For over two millennia mathematicians have been familiar with situations in which they recognized without difficulty that they had in hand an algorithm for a certain class of questions, i.e. the description of a method (that description being finite because it must be completed before picking any one of the questions to apply it to) which constitutes an algorithm as just explained for a given countably infinite class of questions.

Who in 1930 would have guessed that this age-old notion would receive a refinement in the next decade that would put the subject in a new light?

To introduce a little more terminology, for an n-place function $f(a_1, \ldots, a_n)$ $(n \geq 1)$, if an algorithm exists for it, the function f is said to be "computable" or "effectively calculable." The problem of finding an algorithm for it is called its "computation problem." If an algorithm for it does not exist, its computation problem is "unsolvable" and it is "uncomputable."

For an n-place relation or "predicate" $R(a_1, \ldots, a_n)$ $(n \geq 1)$, if an algorithm exists for it, the predicate is said to be "decidable." The problem of finding an algorithm for it is called its "decision problem." If an algorithm does not exist for it, its decision problem is "unsolvable," and it is "undecidable." A relation $R(a_1, \ldots, a_n)$ is decidable exactly if the function (called its "representing function")

$$f(a_1, \ldots, a_n) = \begin{cases} 0 & \text{if } R(a_1, \ldots, a_n) \text{ is true} \\ 1 & \text{if } R(a_1, \ldots, a_n) \text{ is false} \end{cases}$$

is effectively calculable.

We shall concentrate on the cases of functions and predicates with the nonnegative integers (briefly "natural numbers") as the values of the independent variables. Each other case that concerns us can be mapped on such a case, indeed with $n = 1$, by effectively enumerating the class of the n-tuples of arguments for it.

What happened in the period 1934–1937 was that three exact formulations of a class of functions $f(a_1, \ldots, a_n)$ of natural-number variables arose which came to be identified with the effectively calculable functions of those variables — with those for which there are algorithms. These were the "λ-definable" functions of Church and Kleene 1934, the "general recursive" functions of Gödel 1934 adapting an idea of Herbrand (in France), and the "[Turing] computable" functions of Turing (in England) and Post 1936–1937. (For the twentieth century, I indicate the country in which the work was done when it was not the United States.) The λ-definable functions and the Herbrand-Gödel general recursive functions were proved to be the same class of functions by Kleene in 1936, and the Turing computable functions to be the same as the λ-definable functions by Turing in 1937. The proposal to identify this common class with the effectively calculable functions, as first published by Church in 1936 using the first two formulations, has come to be called "Church's thesis." Turing independently and Post (knowing of the work of Church, Kleene, and Gödel) proposed the like for their (essentially similar) formulations in 1936–1937, so we have "Turing's thesis" or the "Church-Turing thesis."

For Church, the conclusion that all the effectively calculable functions are comprised in the class of the λ-definable functions rested on very comprehensive closure properties established for it (and similarly for the general recursive functions). Turing described a kind of machine ("Turing machines") designed to perform equivalents of all the kinds of steps a human computer could perform when confined to following a preassigned finite list of instructions.

That, conversely, all λ-definable, general recursive and Turing computable functions are effectively calculable is evident from the way they are each defined.

Church and Turing each gave at once examples of predicates $R(a_1, \dots, a_n)$ for which, on the basis of their theses, no algorithm can exist; in other words, of unsolvable decision problems. Their first examples were, respectively, from the theory of λ-definability and the theory of Turing computability.

They each proceeded thence to establish the unsolvability of Hilbert's famous Entscheidungsproblem (which Hilbert considered to be the fundamental problem of mathematical logic) for the case of the (restricted or first-order) predicate calculus (engere Funktionenkalkul) of Hilbert and Ackermann's 1928 book. The problem is to decide as to the provability or unprovability of any given formula in the predicate calculus, the solution of which would draw with it the solvability by purely mechanical means of a host of mathematical problems.

That there are uncomputable functions and undecidable relations is indeed immediate from Church's thesis or Turing's thesis without delving into details, simply because each λ-definable, general recursive or Turing computable function has a finite definition in a respective finite symbolism, so there are only countably many definitions of them in that symbolism. So there are only countably many effectively calculable functions (of any number $n \geq 1$ of natural number variables) and hence only countably many decidable predicates. But by Cantor's famous diagonal method of 1874, the set of all number-theoretic functions (indeed of any number $n \geq 1$ of variables, or of ones taking only 0 and 1 as values and thus representing predicates) is uncountable.

The question remaining is to get interesting examples of unsolvable decision (or computation) problems, such as the unsolvability of decision problems that have already come to mind, or the undecidability of predicates of an interesting logical form. Examples of the former were given, as we have said, by Church and Turing. The latter was done by Kleene in 1936, using the formulation in terms of general recursiveness. He constructed a very elementary decidable 3-place predicate $T_1(a, b, x)$ (what he called "primitive recursive") such that the 1-place predicate $(Ex)T_1(a, a, x)$ (and hence also its negation $(x)\overline{T}_1(a, a, x))$ is undecidable. Here "(Ex)" means "there exists an x," and "(x)" means "for all x," the two operations being called "quantifiers."

Should these various results interest mathematicians outside of the area of logic and foundations? Church wrote to me on May 19, 1936, "What I would really like done would be my [unsolvability] results or yours used to prove the unsolvability of some mathematical problems of this order not on their face specially related to logic."

Just this has happened, though the details were drawn from the Turing-Post formulation (not yet in print on May 19, 1936). In 1947 Post, and independently Markov (in Russia), showed on the basis of the Church-Turing-Post thesis that the "word problem for semi-groups" is unsolvable. The "word problem for groups," a celebrated problem for algebraists, who had failed in intensive efforts to solve it, was shown to be unsolvable by Novikov (in Russia) in 1955 (in a 143 page paper). Incidentally, Post in 1943, Markov in 1951, and Smullyan in 1961 gave further characterizations of unsolvability, equivalent to the three we have named.

In 1958, Markov showed the "homeomorphism problem for four-dimensional manifolds" in topology to be unsolvable.

Unsolvability results in real-variable analysis appeared in work of Scarpellini (in Switzerland) in 1963 and of Richardson (in England) in 1966.

The theory of recursive (and non-recursive) functions and predicates developed into a very substantial new discipline. We know from the examples $(Ex)T_1(a, a, x)$ and $(x)\overline{T}_1(a, a, x)$ that applying one quantifier to suitable decidable predicates produces some undecidable predicates. There are of course uncountably many undecidable predicates (as the class of all predicates is uncountable). So there must be some not of the form $(Ex)R(a, x)$ or the form $(x)R(a, x)$ with R decidable. Indeed, we get some more undecidable predicates by taking ones of the forms $(x)(Ey)R(a, x, y)$ and $(Ex)(y)R(a, x, y)$ with R decidable. Continuing thus, Kleene in 1943 and independently Mostowski in 1947 described a hierarchy with the decidable predicates comprising the lowest level. This hierarchy was greatly expanded when quantifiers over number-theoretic functions and functionals of increasing types were considered by Kleene beginning in 1955, using relativized algorithms (after an idea of Turing 1939 USA) in which we allow as inputs values of function variables for arguments arising in the course of applying the algorithm. This development has contacts with descriptive set theory (originated by Borel, Baire, Lebesgue around the turn of the century).

Post in 1944 and 1948 introduced a notion of "degree of unsolvability," first elaborated in a joint paper with Kleene in 1954. Two predicates $R(a)$ and $S(a)$ have the *same degree* of unsolvability if there is an algorithm for $R(a)$ which is relativized to $S(a)$ by allowing inputs of values of $S(a)$ and vice versa (if not vice versa, $R(a)$ is of *lower degree* of unsolvability than $S(a)$).

The collection of the degrees of unsolvability has a complicated structure, which has been investigated in detail.

Is set theory in the mainstream of mathematics? Or is it only a branch of logic and foundations? At any rate set-theorists concerned with the continuum problem must now take into account the work of Gödel in the logical foundations of set theory. In 1938 he obtained the first significant result on the continuum hypothesis since Cantor formulated it in 1878. Gödel used a model to show that the continuum hypothesis is consistent with the quite standard set of axioms ZFC for set theory, originating with Zermelo and Fraenkel (both in Germany) in the period 1904–1922, including Zermelo's axiom of choice (the C). Paul Cohen in 1963–1964 using a different model showed that the negation of continuum hypothesis is also consistent with ZFC. So the continuum hypothesis is independent of ZFC. If the continuum problem is to be solved, it must be with the help of new acceptable axioms; or two set theories must be established with alternative acceptable axioms (like Euclidean and non-Euclidean geometry) diverging from each other on the continuum hypothesis.

An extensive theory of models has now developed in mathematical logic especially from the late 1940s and early 1950s.

Gödel's famous incompleteness theorem of 1931 (found while he was in Austria), with the generalization of it by Kleene in 1943 which I describe next, has the significance for mathematicians that, when they want to answer particular questions, they should consider the collection of the axioms and principles of inference they are prepared to use. To make this exact, this means collecting them into a "formal system" with an agreed mathematical logic. Except for quite trivial domains, any system embodying stated methods thus fixed in advance as a formal system will be inadequate for answering some questions: precisely this is the case for any formal system that is correct (in the sense called "ω-consistent") and adequate for some elementary number theory. In such a system not all of the formulas expressing true values of the particular predicate $(x)\overline{T}_1(a,a,x)$ mentioned above will be provable. (Thus the theory of this one predicate provides inexhaustible scope for mathematical ingenuity, as contrasted with patience no matter how great in applying already formulated methods.) For, from the nature of a formal system adequate for a modicum of elementary number theory, there should be an algorithm for finding a formula in the system expressing $(x)\overline{T}_1(a,a,x)$ for a given a, and another algorithm for recognizing when a finite sequence of formulas in the system is a proof (of its last formula). Also in the systems considered, the negation of that formula will be provable whenever that formula is false (working from an example of a number x for which $T_1(a,a,x)$ is true). Now, if the system would prove the formula expressing $(x)\overline{T}_1(a,a,x)$ for every a for which it is true, we would have an algorithm for deciding the predicate $(x)\overline{T}_1(a,a,x)$ that would consist in searching through the proofs in the system looking for a proof of the formula expressing $(x)\overline{T}_1(a,a,x)$ or a proof of the formula expressing its negation (equivalent to $(Ex)T_1(a,a,x)$). This would contradict the above-mentioned undecidability of $(x)\overline{T}_1(a,a,x)$.

I have refrained from citing papers, as a reader of this article not already versed in logic and foundations would have trouble reading most of them in isolation. I venture to mention two books which are reasonably self-contained and expound many details on the matters touched upon in the present article: S. C. Kleene, *Introduction to Metamathematics*, North-Holland Publishing Co., Amsterdam, 1952, 9th reprint 1988; C. C. Chang and H. J. Keisler, *Model Theory*, North-Holland Publishing Co., Amsterdam, 1973, 2nd edition, 2nd reprint 1981.