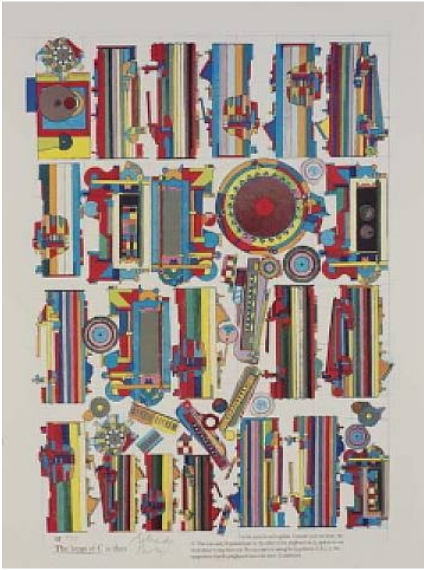
Turing Centenary Lecture



P.D.Welch

University of Bristol

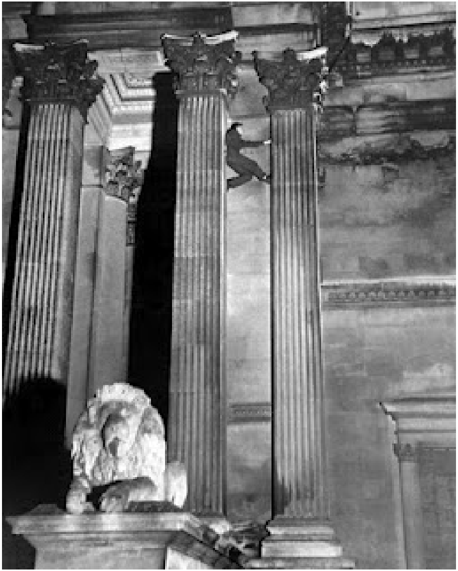
Visiting Research Fellow,

Isaac Newton Institute

Early Life

King’s College 1931

King’s College 1931



Hardy

Eddington

He attended Eddington's lectures on the Distribution of Measurements in Scientific Experiments.

• This led Turing to a rediscovery of the Central Limit Theorem in Feb. 1934.

He attended Eddington's lectures on the Distribution of Measurements in Scientific Experiments.

• This led Turing to a rediscovery of the Central Limit Theorem in Feb. 1934.

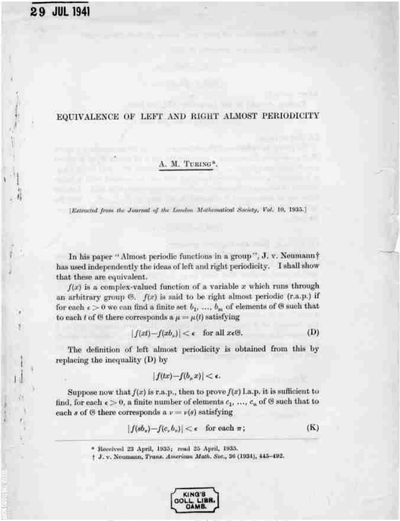
• However this had already been proven in a similar form by Lindeberg in 1922.

• Fellowship Dissertation:
On the Gaussian Error Function

• Accepted 16 March 1936. This was £300 p.a. Age: 22.

PHILIP HALL

First start in group theory[1]



EQUIVALENCE OF LEFT AND RIGHT ALMOST PERIODICITY

A. M. Turing*.

[*Extracted from the Journal of the London Mathematical Society, Vol.* 10, 1935.]

In his paper "Almost periodic functions in a group", J. v. Neumann† has used independently the ideas of left and right periodicity. I shall show that these are equivalent.

$f(x)$ is a complex-valued function of a variable x which runs through an arbitrary group $\mathfrak{G}$. $f(x)$ is said to be right almost periodic (r.a.p.) if for each $\epsilon > 0$ we can find a finite set $b_1, \ldots, b_m$ of elements of $\mathfrak{G}$ such that to each t of $\mathfrak{G}$ there corresponds a $\mu = \mu(t)$ satisfying

$$|f(xt) - f(xb_\mu)| < \epsilon \quad \text{for all } x \in \mathfrak{G}. \qquad \text{(D)}$$

The definition of left almost periodicity is obtained from this by replacing the inequality (D) by

$$|f(tx) - f(b_\mu x)| < \epsilon.$$

Suppose now that $f(x)$ is r.a.p., then to prove $f(x)$ l.a.p. it is sufficient to find, for each $\epsilon > 0$, a finite number of elements $c_1, \ldots, c_n$ of $\mathfrak{G}$ such that to each s of $\mathfrak{G}$ there corresponds a $\nu = \nu(s)$ satisfying

$$|f(sb_\pi) - f(c_\nu b_\pi)| < \epsilon \quad \text{for each } \pi; \qquad \text{(K)}$$

* Received 23 April, 1935; read 25 April, 1935.

† J. v. Neumann, *Trans. American Math. Soc.*, 36 (1934), 445–492.

[1] *Equivalence of left and right almost periodicity" J. of the London Math. Society,10, 1935.*

Max Newman

Hilbert's grave

Hilbert's program and the *Entscheidungsproblem*.

- (I Completeness) His dictum concerning the belief that any mathematical problem was in principle solvable, can be restated as the belief that mathematics was *complete*: that is, given any properly formulated mathematical proposition P, either a proof of P could be found, or a disproof.

Hilbert's program and the *Entscheidungsproblem*.

• (I Completeness) His dictum concerning the belief that any mathematical problem was in principle solvable, can be restated as the belief that mathematics was *complete*: that is, given any properly formulated mathematical proposition P, either a proof of P could be found, or a disproof.

• (II Consistency) the question of *consistency*: given a set of axioms for, say, arithmetic, such as the Dedekind-Peano axioms,PA, could it be shown that no proof of a contradiction can possibly arise? Hilbert stringently wanted a proof of consistency that was finitary, that made no appeal to infinite objects or methods.

Hilbert's program and the *Entscheidungsproblem*.

- (I Completeness) His dictum concerning the belief that any mathematical problem was in principle solvable, can be restated as the belief that mathematics was *complete*: that is, given any properly formulated mathematical proposition P, either a proof of P could be found, or a disproof.

- (II Consistency) the question of *consistency*: given a set of axioms for, say, arithmetic, such as the Dedekind-Peano axioms,PA, could it be shown that no proof of a contradiction can possibly arise? Hilbert stringently wanted a proof of consistency that was finitary, that made no appeal to infinite objects or methods.

- (III Decidability - the *Entscheidungsproblem*) Could there be a finitary process or algorithm that would *decide* for any such P whether it was derivable from axioms or not?

Hilbert's program and the *Entscheidungsproblem*.

• (I Completeness) His dictum concerning the belief that any mathematical problem was in principle solvable, can be restated as the belief that mathematics was *complete*: that is, given any properly formulated mathematical proposition P, either a proof of P could be found, or a disproof.

• (II Consistency) the question of *consistency*: given a set of axioms for, say, arithmetic, such as the Dedekind-Peano axioms,*PA*, could it be shown that no proof of a contradiction can possibly arise? Hilbert stringently wanted a proof of consistency that was finitary, that made no appeal to infinite objects or methods.

• (III Decidability - the *Entscheidungsproblem*) Could there be a finitary process or algorithm that would *decide* for any such P whether it was derivable from axioms or not?

• There was both hope (from the Göttingen group) that the *Entscheidungsproblem* was soluble . . .

Skepticism

. . . and from others that it was not:

[2] *"Mathematical Proof"* , Mind, 1929.
[3] *"Zur Hilbertsche Beweistheorie"*, Math. Z., 1927.

Skepticism

. . . and from others that it was not:

• Hardy: *"There is of course no such theorem [that there is a positive solution to the* Entscheidungsproblem *holds] and this is very fortunate, since if there were we should have a mechanical set of rules for the solution of all mathematical problems, and our activities as mathematicians would come to an end."*[2]

[2] *"Mathematical Proof"* , Mind, 1929.
[3] *"Zur Hilbertsche Beweistheorie"*, Math. Z., 1927.

Skepticism

. . . and from others that it was not:

• Hardy: *“There is of course no such theorem [that there is a positive solution to the* Entscheidungsproblem *holds] and this is very fortunate, since if there were we should have a mechanical set of rules for the solution of all mathematical problems, and our activities as mathematicians would come to an end.”*[2]

• von Neumann: *“ When undecidability fails, then mathematics as it is understood today ceases to exist; in its place there would be an absolutely mechanical prescription with whose help one could decide whether any given sentence is provable or not.”*[3]

---

[2] *“Mathematical Proof”* , Mind, 1929.

[3] *“Zur Hilbertsche Beweistheorie”*, Math. Z., 1927.

Gödel and Incompleteness

Theorem (Gödel-Rosser First Incompleteness Theorem - 1931)

For any theory T containing a moderate amount of arithmetical strength, with T having an effectively given list of axioms, then: if T is consistent, then it is incomplete, that is for some proposition neither $T \vdash P$ nor $T \vdash \neg P$.

Gödel and Incompleteness

Theorem (Gödel-Rosser First Incompleteness Theorem - 1931)

For any theory T containing a moderate amount of arithmetical strength, with T having an effectively given list of axioms, then:
if T is consistent, then it is incomplete, that is for some proposition neither $T \vdash P$ nor $T \vdash \neg P$.

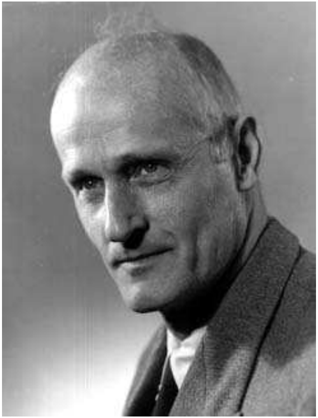
Theorem (Gödel's Second Incompleteness Theorem - 1931)

For any consistent T as above, containing the axioms of PA, the statement that 'T is consistent' (when formalised as 'Con_T') is an unprovable sentence.
Symbolically: $T \nvdash \mathrm{Con}_T$.

Church and the λ-calculus

A. CHURCH

- The λ-*calculus* - a strict formalism for writing out terms defining a class of functions from base functions and an induction scheme.



S.C.KLEENE

• They used the term *"effectively calculable functions"*: the class of functions that could be calculated in the informal sense of effective procedure or algorithm.

1934 - Princeton

- **Church's Thesis (1934 - First version, unpublished)** ***The effectively calculable functions coincide with the λ-definable functions.***

1934 - Princeton

- **Church's Thesis (1934 - First version, unpublished)** ***The effectively calculable functions coincide with the λ-definable functions*****.**

Gödel:

1934 - Princeton

- **Church's Thesis (1934 - First version, unpublished)** ***The effectively calculable functions coincide with the λ-definable functions.***

Gödel: “thoroughly unsatisfactory”.

The switch to Herbrand-Gödel general recursive functions

- Gödel introduced the *Herbrand-Gödel general recursive functions* (1934).

Kleene:
"I myself, perhaps unduly influenced by rather chilly receptions from audiences around 1933—35 to disquisitions on –λ-definability, chose, after [Herbrand-Gödel] general recursiveness had appeared, to put my work in that format. . . . "

Preliminary solutions to the *Entscheidungsproblem*

• By 1935 Church could show that there was no λ formula "*A conv B*" iff A and B were convertible to each other within the λ-calculus.

[4] *"An Unsolvable problem of elementary number theory"*, 58, J. of AMS 1936

Preliminary solutions to the *Entscheidungsproblem*

• By 1935 Church could show that there was no λ formula "*A conv B*" iff A and B were convertible to each other within the λ-calculus.

• Moreover, the λ definable functions were co-extensive with the general recursive functions.

[4] *"An Unsolvable problem of elementary number theory"*, 58, J. of AMS 1936

Preliminary solutions to the *Entscheidungsproblem*

• By 1935 Church could show that there was no λ formula "*A conv B*" iff A and B were convertible to each other within the λ-calculus.

• Moreover, the λ definable functions were co-extensive with the general recursive functions.

• There was therefore a problem in number theory that could not be solved using general recursive functions. This was published by Church [4]

[4] *"An Unsolvable problem of elementary number theory"*, 58, J. of AMS 1936

Preliminary solutions to the *Entscheidungsproblem*

• By 1935 Church could show that there was no λ formula "*A conv B*" iff A and B were convertible to each other within the λ-calculus.

• Moreover, the λ definable functions were co-extensive with the general recursive functions.

• There was therefore a problem in number theory that could not be solved using general recursive functions. This was published by Church [4]

Another thesis was formulated:

• **Church's Thesis (1936 - second version)** ***The effectively calculable functions coincide with the [H-G] general recursive functions.***

---

[4] *"An Unsolvable problem of elementary number theory"*, 58, J. of AMS 1936

Preliminary solutions to the *Entscheidungsproblem*

• By 1935 Church could show that there was no λ formula "*A conv B*" iff A and B were convertible to each other within the λ-calculus.

• Moreover, the λ definable functions were co-extensive with the general recursive functions.

• There was therefore a problem in number theory that could not be solved using general recursive functions. This was published by Church [4]

Another thesis was formulated:

• **Church's Thesis (1936 - second version)** ***The effectively calculable functions coincide with the [H-G] general recursive functions.***

Gödel: he remained unconvinced.

---

[4] *"An Unsolvable problem of elementary number theory"*, 58, J. of AMS 1936

On Computable Numbers

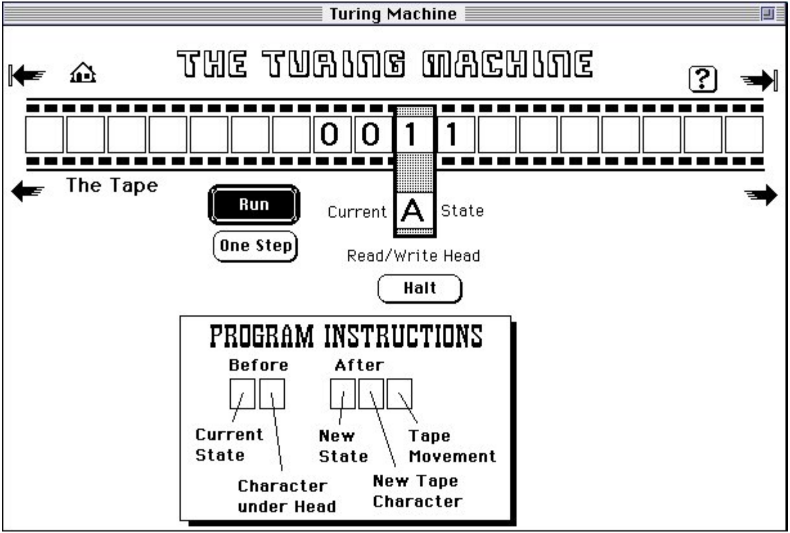
ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. TURING.

[Received 28 May, 1936.—Read 12 November, 1936.]

The "computable" numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable *numbers*, it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers

Turing's Analysis: Section 1



"... these operations include all those which are used in the computation of a number."

Section 2

" If at any each stage the motion of the machine is completely *determined by the configuration, we shall call the machine an "automatic" or a-machine."*

Section 2

“ *If at any each stage the motion of the machine is* completely *determined by the configuration, we shall call the machine an “automatic” or a-machine.”*

“For some purposes we may use machines whose motion is only partly determined. When such a machine reaches one of these ambiguous configurations, it cannot go on until some arbitrary choice has been made ...”

Section 9 “The extent of the computable numbers”

- A complete analysis of human computation in terms of finiteness of the human acts of calculation broken-down into discrete, simple, and locally determined steps.

[5] *“The confluence of ideas in 1936”*, in “The Universal Turing Machine”, Ed R. Herkel, OUP, 1988.

Section 9 “The extent of the computable numbers”

• A complete analysis of human computation in terms of finiteness of the human acts of calculation broken-down into discrete, simple, and locally determined steps.

• It is important to see that this analysis should be taken *prior to* the machine’s description. He had asked:

> “What are the possible *processes* which can be carried out in computing a real number” *[My emphasis]*.

---

[5] *“The confluence of ideas in 1936”*, in “The Universal Turing Machine”, Ed R. Herkel, OUP, 1988.

Section 9 "The extent of the computable numbers"

• A complete analysis of human computation in terms of finiteness of the human acts of calculation broken-down into discrete, simple, and locally determined steps.

• It is important to see that this analysis should be taken *prior to* the machine's description. He had asked:

> "What are the possible *processes* which can be carried out in computing a real number" *[My emphasis]*.

Turing's Thesis: Anything that is humanly calculable is computable by a Turing machine.

---

[5] *"The confluence of ideas in 1936"*, in "The Universal Turing Machine", Ed R. Herkel, OUP, 1988.

Section 9 “The extent of the computable numbers”

- A complete analysis of human computation in terms of finiteness of the human acts of calculation broken-down into discrete, simple, and locally determined steps.

- It is important to see that this analysis should be taken *prior to* the machine’s description. He had asked:

> “What are the possible *processes* which can be carried out in computing a real number” *[My emphasis]*.

Turing’s Thesis: Anything that is humanly calculable is computable by a Turing machine.

- According to Gandy[5] Turing has in fact proved a *theorem* albeit one with unusual subject matter.

---

[5] *“The confluence of ideas in 1936”*, in “The Universal Turing Machine”, Ed R. Herkel, OUP, 1988.

What did Turing achieve - besides specifying a universal machine?

- Turing provides a philosophical paradigm when defining “effectively calculable.”

What did Turing achieve - besides specifying a universal machine?

- Turing provides a philosophical paradigm when defining “effectively calculable.”

- He also makes completely precise what is a ‘formal system’.

- It is because of T’s analysis that we can assert statements such as “Hilbert’s 10th problem is an undecidable”.

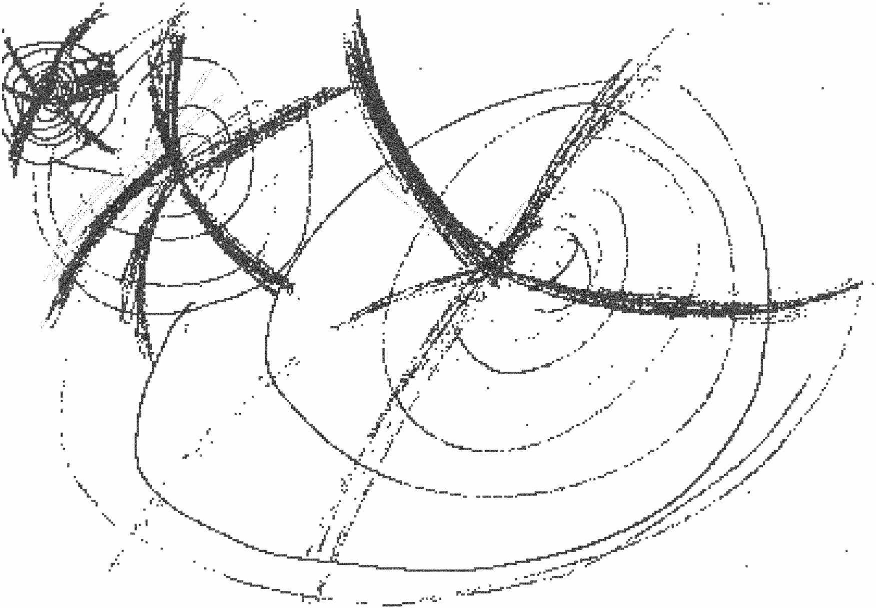
What did Turing achieve - besides specifying a universal machine?

• Turing provides a philosophical paradigm when defining "effectively calculable."

• He also makes completely precise what is a 'formal system'.

• It is because of T's analysis that we can assert statements such as "Hilbert's 10th problem is an undecidable".

In the final 4 pages he gives his solution to the *Entscheidungsproblem*. He proves that there is no machine that will decide of any formula φ of the predicate calculus whether it is derivable or not.



Gödel again:[6]

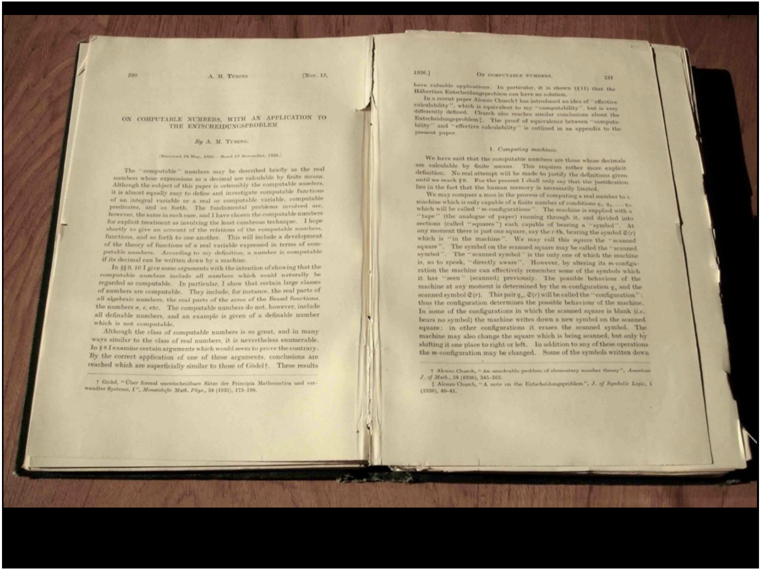
> *"When I first published my paper about undecidable propositions the result could not be pronounced in this generality, because for the notions of mechanical procedure and of formal system no mathematically satisfactory definition had been given at that time. . . . The essential point is to define what a procedure is."*

> *"That this really is the correct definition of mechanical computability was established beyond any doubt by Turing."*

---

[6]From a Lecture in the *Nachlass* Vol III, p166-168.

On Computable Numbers-IAS



ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. Turing.

[Received 28 May, 1936.—Read 12 November, 1936.]

The "computable" numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable *numbers*, it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers for explicit treatment as involving the least cumbrous technique. I hope shortly to give an account of the relations of the computable numbers, functions, and so forth to one another. This will include a development of the theory of functions of a real variable expressed in terms of computable numbers. According to my definition, a number is computable if its decimal can be written down by a machine.

In §§ 9, 10 I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be regarded as computable. In particular, I show that certain large classes of numbers are computable. They include, for instance, the real parts of all algebraic numbers, the real parts of the zeros of the Bessel functions, the numbers π, e, etc. The computable numbers do not, however, include all definable numbers, and an example is given of a definable number which is not computable.

Although the class of computable numbers is so great, and in many ways similar to the class of real numbers, it is nevertheless enumerable. In § 8 I examine certain arguments which would seem to prove the contrary. By the correct application of one of these arguments, conclusions are reached which are superficially similar to those of Gödel†. These results

† Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I", *Monatshefte Math. Phys.*, 38 (1931), 173-198.

have valuable applications. In particular, it is shown (§ 11) that the Hilbertian Entscheidungsproblem can have no solution.

In a recent paper Alonzo Church† has introduced an idea of "effective calculability", which is equivalent to my "computability", but is very differently defined. Church also reaches similar conclusions about the Entscheidungsproblem‡. The proof of equivalence between "computability" and "effective calculability" is outlined in an appendix to the present paper.

1. *Computing machines.*

We have said that the computable numbers are those whose decimals are calculable by finite means. This requires rather more explicit definition. No real attempt will be made to justify the definitions given until we reach § 9. For the present I shall only say that the justification lies in the fact that the human memory is necessarily limited.

We may compare a man in the process of computing a real number to a machine which is only capable of a finite number of conditions $q_1, q_2, \ldots, q_R$ which will be called "m-configurations". The machine is supplied with a "tape" (the analogue of paper) running through it, and divided into sections (called "squares") each capable of bearing a "symbol". At any moment there is just one square, say the r-th, bearing the symbol $\mathfrak{S}(r)$ which is "in the machine". We may call this square the "scanned square". The symbol on the scanned square may be called the "scanned symbol". The "scanned symbol" is the only one of which the machine is, so to speak, "directly aware". However, by altering its m-configuration the machine can effectively remember some of the symbols which it has "seen" (scanned) previously. The possible behaviour of the machine at any moment is determined by the m-configuration q_n and the scanned symbol $\mathfrak{S}(r)$. This pair q_n, $\mathfrak{S}(r)$ will be called the "configuration": thus the configuration determines the possible behaviour of the machine. In some of the configurations in which the scanned square is blank (*i.e.* bears no symbol) the machine writes down a new symbol on the scanned square: in other configurations it erases the scanned symbol. The machine may also change the square which is being scanned, but only by shifting it one place to right or left. In addition to any of these operations the m-configuration may be changed. Some of the symbols written down

† Alonzo Church, "An unsolvable problem of elementary number theory", *American J. of Math.*, 58 (1936), 345-363.

‡ Alonzo Church, "A note on the Entscheidungsproblem", *J. of Symbolic Logic*, 1 (1936), 40-41.

Turing's *"Ordinal logics"*

SYSTEMS OF LOGIC BASED ON ORDINALS†

By A. M. TURING.

[Received 31 May, 1938.—Read 16 June, 1938.]

Turing's *"Ordinal logics"*

Let T_0 be *PA*;

$$T_1 : T_0 + \mathrm{Con}(T_0)$$

where "$\mathrm{Con}(T_0)$" some expression arising from the Incompleteness Theorems expressing that "T_0 is a consistent system"

$$T_{k+1} : T_k + \mathrm{Con}(T_k) \text{ for } k < \omega \text{ , and then:}$$

$$T_\omega = \bigcup_{k<\omega} T_k.$$

Turing's *"Ordinal logics"*

Let T_0 be *PA*;

$$T_1 : T_0 + \mathrm{Con}(T_0)$$

where "$\mathrm{Con}(T_0)$" some expression arising from the Incompleteness Theorems expressing that "T_0 is a consistent system"

$$T_{k+1} : T_k + \mathrm{Con}(T_k) \text{ for } k < \omega \text{ , and then:}$$

$$T_\omega = \bigcup_{k<\omega} T_k .$$

Presumably we may continue:

$$T_{\omega+1} = T_\omega + \mathrm{Con}(T_\omega) \text{ etc.}$$

Ordinal logic continued

Question: Can it be that for any arithmetical problem A there might be an ordinal α so that T_α proves A or $\neg A$?

Ordinal logic continued

Question: Can it be that for any arithmetical problem A there might be an ordinal α so that T_α proves A or $\neg A$?

• Let $\varphi_0(v_0)$ define the set of gödel code numbers n of instances of the axiom set $T_0 = \text{PA}$.

Ordinal logic continued

Question: Can it be that for any arithmetical problem A there might be an ordinal α so that T_α proves A or $\neg A$?

• Let $\varphi_0(v_0)$ define the set of gödel code numbers n of instances of the axiom set $T_0 = \mathrm{PA}$.

• Then set

$$\varphi_{k+1}(\overline{n}) \longleftrightarrow \varphi_k(\overline{n}) \vee \overline{n} = \mathrm{Con}(\varphi_k)$$

where $\mathrm{Con}(\varphi_k)$ expresses in a Gödelian fashion the consistency of the axiom set defined by φ_k.

Ordinal logic continued

Question: Can it be that for any arithmetical problem A there might be an ordinal α so that T_α proves A or $\neg A$?

• Let $\varphi_0(v_0)$ define the set of gödel code numbers n of instances of the axiom set $T_0 = \mathrm{PA}$.

• Then set

$$\varphi_{k+1}(\overline{n}) \longleftrightarrow \varphi_k(\overline{n}) \vee \overline{n} = \mathrm{Con}(\varphi_k)$$

where $\mathrm{Con}(\varphi_k)$ expresses in a Gödelian fashion the consistency of the axiom set defined by φ_k.

• But φ_ω ??

Notations

- The problem can be solved really only with numerical *notations for ordinals*.

- The set of notations $\mathcal{O} \subset \mathbb{N}$ is essentially a tree order with

$$n <_{\mathcal{O}} m \leftrightarrow |n| < |m|.$$

- If n is the immediate $<_{\mathcal{O}}$ predecessor of m, then $T_m = T_n + Con(\varphi_n)$.

- If $m \in \mathcal{O}$ and $|m|$ a limit ordinal given by a total TM $\{e\}$ then

$$T_m = \bigcup_k T_{\{e\}(k)}.$$

(But we end up needing to assign theories T_b for all numbers b.)

Theorem (Turing's Completeness Theorem)

There is an assignment $\psi \to b(\psi)$ *so that for any true* $\forall$ *sentence of arithmetic,* ψ*,* $b = b(\psi) \in \mathcal{O}$ *with* $|b| = \omega + 1$*, so that* $T_b \vdash \psi$*.*

Thus we may for any true ψ find a path through $\mathcal{O}$ of length $\omega + 1$,

$$T = T_0, T_1, \ldots, T_{\omega+1} = T_b$$

with $T_b \vdash \psi$.

Theorem (Turing's Completeness Theorem)

There is an assignment $\psi \to b(\psi)$ *so that for any true* $\forall$ *sentence of arithmetic,* ψ*,* $b = b(\psi) \in \mathcal{O}$ *with* $|b| = \omega + 1$*, so that* $T_b \vdash \psi$*.*

Thus we may for any true ψ find a path through $\mathcal{O}$ of length $\omega + 1$,

$$T = T_0, T_1, \ldots, T_{\omega+1} = T_b$$

with $T_b \vdash \psi$.

• There is a trick here: what one does is construct for *any* $\forall$ sentence ψ an extension $T_{b(\psi)}$ proving ψ

Theorem (Turing's Completeness Theorem)

There is an assignment $\psi \to b(\psi)$ *so that* **for any true** $\forall$ *sentence of arithmetic,* ψ*, there is a* $b = b(\psi) \in \mathcal{O}$ *with* $|b| = \omega + 1$*, so that* $T_b \vdash \psi$*.*

Thus we may for any true ψ find a path through $\mathcal{O}$ of length $\omega + 1$,

$$T = T_0, T_1, \ldots, T_{\omega+1} = T_b$$

with $T_b \vdash \psi$.

• There is a trick here: what one does is construct for *any* $\forall$ sentence ψ an extension $T_{b(\psi)}$ proving ψ.

Then:

if ψ *is true* we deduce that $b \in \mathcal{O}$ (and is a notation for $\omega + 1$) with $T_{b(\psi)}$ a consistency extension proving ψ.

Theorem (Turing's Completeness Theorem)

There is an assignment $\psi \to b(\psi)$ so that **for any true** *$\forall$ sentence of arithmetic, ψ, there is a $b = b(\psi) \in \mathcal{O}$ with $|b| = \omega + 1$, so that $T_b \vdash \psi$.*

Thus we may for any true ψ find a path through $\mathcal{O}$ of length $\omega + 1$,

$$T = T_0, T_1, \ldots, T_{\omega+1} = T_b$$

with $T_b \vdash \psi$.

• There is a trick here: what one does is construct for *any* $\forall$ sentence ψ an extension $T_{b(\psi)}$ proving ψ.

Then:

if ψ is true we deduce that $b \in \mathcal{O}$ (and is a notation for $\omega + 1$) with $T_{b(\psi)}$ a consistency extension proving ψ.
if ψ is false $T_{b(\psi)}$ is inconsistent.

Theorem (Turing's Completeness Theorem)

There is an assignment $\psi \to b(\psi)$ *so that* **for any true** $\forall$ *sentence of arithmetic,* ψ*, there is a* $b = b(\psi) \in \mathcal{O}$ *with* $|b| = \omega + 1$*, so that* $T_b \vdash \psi$*.*

Thus we may for any true ψ find a path through $\mathcal{O}$ of length $\omega + 1$,

$$T = T_0, T_1, \ldots, T_{\omega+1} = T_b$$

with $T_b \vdash \psi$.

• There is a trick here: what one does is construct for *any* $\forall$ sentence ψ an extension $T_{b(\psi)}$ proving ψ.

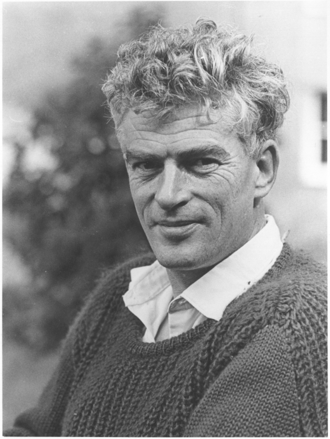
Then:

if ψ *is true* we deduce that $b \in \mathcal{O}$ (and is a notation for $\omega + 1$) with $T_{b(\psi)}$ a consistency extension proving ψ.
if ψ *is false* $T_{b(\psi)}$ is inconsistent.

• In general it is harder to answer ?$b \in \mathcal{O}$? than the original $\forall$ question, and so we have gained no new arithmetical knowledge.

"This is only a foretaste of what is to come, and only the shadow of what is going to be. We have to have some experience with the machine before we really know its capabilities. It may take years before we settle down to the new possibilities, but I do not see why it should not enter any of the fields normally covered by the human intellect and eventually compete on equal terms." (Press Interview with *The Times* June 1949)

Robin Gandy



In Memoriam: Robin Gandy 1919 - 1995