We might imagine a machine where we put in axioms at one end and take out theorems at the other, like that legendary machine in Chicago where pigs go in alive and come out transformed into hams and sausages.

-Henri Poincaré
Alan Turing

On Computable Numbers, with an Application to the Entscheidungsproblem
TAPELEABLE NUMBERS, WITH AN APPLICATION TO THE

ENTScheidungsprobleM

By A. M. Turing.

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

The numbers may be described briefly as the real numbers as a decimal are calculable by finite means. This paper is ostensibly the computable numbers.

To define and investigate computable functions of a real or computable variable, computable

and so forth. The fundamental problems involved are,

over, 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 \( \pi, e \), 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\(^\dagger\). These results

Alan Turing

On Computable Numbers, with an Application to the Entscheidungsproblem
What does “compute” mean?

What can we compute?

Was it computed?
$q_0$
$q_0$
$q_0 \rightarrow q_7$
$q_0 \rightarrow q_7$
$q_7 \rightarrow q_3$
5-Tuple TM

\( Q \) is the set of states \((q_0, q_1, q_2, \ldots q_n)\)

\( q_0 \) is the initial state

\( q_{\text{halt}} \) stops the machine

\( \Gamma \) is the alphabet \(\{0, 1, _\}\) or \(\{a, b, c, \ldots z, _\}\)

\( \delta \) is the transition function \((Q \times \Gamma \rightarrow \Gamma \times \{L, R\} \times Q)\)
<table>
<thead>
<tr>
<th></th>
<th>$\Gamma$</th>
<th>1</th>
<th>0</th>
<th>_</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$q_0$</td>
<td>$P_0, R, q_1$</td>
<td>$P_0, R, q_1$</td>
<td>$P_1, R, q_2$</td>
<td></td>
</tr>
<tr>
<td>$q_1$</td>
<td>$P_1, L, q_1$</td>
<td>$P_0, L, q_2$</td>
<td>$P_0, R, q_2$</td>
<td></td>
</tr>
<tr>
<td>$q_2$</td>
<td>$P_0, R, q_1$</td>
<td>$P_0, L, q_1$</td>
<td>$P_0, R, q_1$</td>
<td></td>
</tr>
<tr>
<td>$Q$</td>
<td>$\Gamma$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>$q_0$</td>
<td>$P_0, R, q_1$</td>
<td>$P_0, R, q_1$</td>
<td>$P_1, R, q_2$</td>
<td></td>
</tr>
<tr>
<td>$q_1$</td>
<td>$q_{halt}$</td>
<td>$P_0, L, q_2$</td>
<td>$q_{halt}$</td>
<td></td>
</tr>
<tr>
<td>$q_2$</td>
<td>$P_0, R, q_1$</td>
<td>$P_0, L, q_1$</td>
<td>$P_0, R, q_1$</td>
<td></td>
</tr>
</tbody>
</table>
\[
\begin{array}{ccc}
Q & \Gamma & \text{Operation} & \text{Final} \\
q_0 & 0 & P0, R & q_1 \\
q_0 & 1 & P0, R & q_1 \\
q_0 & - & P1, R & q_2 \\
q_1 & 0 & P1, L & q_1 \\
q_1 & 1 & P0, L & q_2 \\
q_1 & - & P0, R & q_2 \\
q_2 & 0 & P0, R & q_1 \\
q_2 & 1 & P0, L & q_1 \\
q_2 & - & P0, R & q_1 \\
\end{array}
\]
<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\Gamma$</th>
<th>Operation</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>0</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>_</td>
<td>P1, R</td>
<td>$q_2$</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>P1, L</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>P0, L</td>
<td>$q_2$</td>
</tr>
<tr>
<td></td>
<td>_</td>
<td>P0, R</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>0</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>P0, L</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>_</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>0</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>P0, L</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>_</td>
<td>P0, R</td>
<td>$q_1$</td>
</tr>
</tbody>
</table>
\[
\begin{array}{c|c|c|c}
Q & \Gamma & \text{Operation} & \text{Final} \\
\hline
q_0 & 0, 1 & P0, R & q_1 \\
\hline
q_1 & 0 & P1, R & q_2 \\
& 1 & P1, L & q_1 \\
\hline
q_2 & 0 & P0, R & q_1 \\
& 1 & P0, L & q_1 \\
\end{array}
\]
δ is the transition function \((Q \times \Gamma \to \Gamma \times \{L, R\} \times Q)\)
What does “compute” mean?

What *can* we compute?

Was it computed?
What does “compute” mean?

What *can* we compute?

Was it computed?
What can this machine compute?

Every number that is calculable by finite means!
\[
\frac{1}{2} = 0.5 = 0.1 \approx 0.10000000000000000000
\]
\[ Q \xrightarrow{\Gamma} \begin{array}{c} P1, R \\ q_1 \end{array} \quad \begin{array}{c} P0, R \\ q_1 \end{array} \]
$q_0$
$q_0 \rightarrow q_1$
$q_1$
\[ \frac{1}{2} = .5 = .1 = .10000000000000000000 \]
\[
\frac{1}{2} = .5 = \frac{1}{1} = .1 = \frac{1}{10000000000000000000}
\]
\[
\frac{1}{64} = .015625 = .00000100000000000000
\]
\[
q_0
\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000 \\
\frac{1}{64} = .015625 = .00000100000000000000
\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000
\]

\[
\frac{1}{64} = .015625 = .00000100000000000000
\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000
\]

\[
\frac{1}{64} = .015625 = .00000100000000000000
\]

\[q_3\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000
\]

\[
\frac{1}{64} = .015625 = .00000100000000000000
\]

\[
q_4
\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000
\]
\[
\frac{1}{64} = .015625 = .00000100000000000000
\]

\[q_5\]
\[
\frac{1}{2} = .5 = .1 = .10000000000000000000
\]

\[
\frac{1}{64} = .015625 = .00000100000000000000
\]
<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\Gamma$</th>
<th>Operation</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>_</td>
<td>P0, R</td>
<td>$q_3$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>_</td>
<td>P_, R</td>
<td>$q_1$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>_</td>
<td>P1, R</td>
<td>$q_4$</td>
</tr>
<tr>
<td>$q_4$</td>
<td>_</td>
<td>P_, R</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>_</td>
<td>P0, R</td>
<td>$q_3$</td>
</tr>
<tr>
<td>States</td>
<td>Operation</td>
<td>Final</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-----------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>$q_0$</td>
<td>P0, R, $P_-$, R</td>
<td>$q_1$</td>
<td></td>
</tr>
<tr>
<td>$q_1$</td>
<td>P1, R, $P_-$, R</td>
<td>$q_2$</td>
<td></td>
</tr>
<tr>
<td>$q_2$</td>
<td>P0, R, $P_-$, R</td>
<td>$q_1$</td>
<td></td>
</tr>
</tbody>
</table>
\[ Q \]

\[ \Gamma \]

<table>
<thead>
<tr>
<th>Operation</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0, R, R</td>
<td>( q_1 )</td>
</tr>
<tr>
<td>P1, R, R</td>
<td>( q_2 )</td>
</tr>
<tr>
<td>P0, R, R</td>
<td>( q_1 )</td>
</tr>
</tbody>
</table>
\begin{align*}
Q & \quad \Gamma & \quad \text{Operation} & \quad \text{Final} \\
q_0 & 0 & \text{P0, R} & q_1 \\
1 & \text{P1, R} & q_1
\end{align*}
\[ q_0 \]

| 0 | 1 | 0 |
$q_1$
\[ q_3 \]

-0-0-0-0-
$q_1 \quad \rightarrow \quad 0 \quad \rightarrow \quad 1 \quad \rightarrow \quad 0$
\[ Q \]

<table>
<thead>
<tr>
<th>Operation</th>
<th>Final</th>
</tr>
</thead>
</table>

\[ q₀ \]
<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\Gamma$</th>
<th>Operation</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1$</td>
<td>1</td>
<td>$R, Px, L, L, L$</td>
<td>$q_1$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>0</td>
<td>$R, R$</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>0,1</td>
<td>$P_1, L$</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>_</td>
<td>$P_1, L$</td>
<td>$q_3$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>x</td>
<td>$P__ , R$</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>$\emptyset$</td>
<td>R</td>
<td>$q_4$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>_</td>
<td>L, L</td>
<td>$q_3$</td>
</tr>
<tr>
<td>$q_4$</td>
<td>0,1</td>
<td>$R, R$</td>
<td>$q_4$</td>
</tr>
<tr>
<td>$q_4$</td>
<td>_</td>
<td>$P_0, L, L$</td>
<td>$q_1$</td>
</tr>
</tbody>
</table>
.001011101111011111011111011111101111
\( \delta \) is the transition function \((Q \times \Gamma \rightarrow \Gamma \times \{L, R\} \times Q)\)
\[ q_0 \quad 1 \quad 1 \quad 1 \quad + \quad 1 \quad 1 \quad \cdots \]

\[ 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad \cdots \]

\[ q_{\text{halt}} \]

The diagram shows a transition from state \( q_0 \) to state \( q_{\text{halt}} \).
\[ q_3 \]
What can this machine compute?

Every number that is calculable by finite means!
What can this machine compute?

Everything that falls under our understanding of “compute”!
Alonzo Church

*The Lambda Calculus*
\((\lambda x \lambda y \text{IsFactor}(x, y) \ (9))(3)\)

\(\lambda y \text{IsFactor}(9, y) \ (3)\)

\text{IsFactor}(9, 3)

\text{THIS is the definition of “compute”!}
First Order Logic

1.) Easy to determine a valid formula

2.) Formulas have one, unambiguous meaning

   “I know a little Russian.”

3.) Deductions are purely syntactic
0 1 0
0 1 0 1 0 1
First Order Logic

1.) Easy to determine a valid formula

2.) Formulas have one, unambiguous meaning
   “I know a little Russian.”

3.) Deductions are purely syntactic
\[
\frac{1}{3} = 0.3333\ldots = 0.0101010101010101010101\ldots
\]

IsZero(x)
\[ \frac{1}{3} = 0.3333\ldots = 0.01010101010101010101 \]

\text{IsZero}(0) = \text{true}
\[ \frac{1}{3} = .3333\ldots = .01010101010101010101 \]

\text{IsZero}(1) = \text{false}
\[\frac{1}{3} = .3333\ldots = .01010101010101010101\ldots\]

\[\text{IsZero}(2) = \text{true}\]
\[ \frac{1}{3} = 0.3333... = 0.01010101010101010101 \]

\[
\text{IsZero}(3) = \text{false}
\]
\[ \frac{1}{3} = .\overline{3} = .01010101010101010101 \]

\[ \neg \text{IsZero}(x) \]

\[ \neg \neg \text{IsZero}(x) \]
What can this machine compute?

Everything that falls under our understanding of “compute”!
Alan Turing

On Computable Numbers, with an Application to the Entscheidungsproblem