every computable function is in the class. If we accept these arguments, we have our rigorous definition of computable.

2. Functions and Relations

We must first decide what inputs and outputs to allow. For the moment, we will take our inputs and outputs to be natural numbers, i.e., non–negative integers. We agree that number means natural number unless otherwise indicated. Lower case Latin letters represent numbers.

We now describe the functions to which the notion of computability applies. Let ω be the set of numbers. For each k, ω^k is the set of k–tuples of numbers. Thus ω^1 is ω, and ω^0 has just one member, the empty tuple. When it is not necessary to specify k, we write $\vec{x}$ for $x_1,\ldots,x_k$.

A k–ary function is a mapping of a subset of ω^k into ω. We agree that a function is always a k–ary function for some k. We use capital Latin letters (usually F, G, and H) for functions.

A k–ary function is total if its domain is all of ω^k. A 0–ary total function is clearly determined by its value at the empty tuple. We identify it with this value, so that a 0–ary total function is just a number. A 1–ary total function is called a real. (This terminology comes from set theory, where reals are often identified with real numbers. It will lead to no confusion, since we never deal with real numbers.)

A common type of algorithm has as output a yes or no answer to some question about the inputs. Since we want our outputs to be numbers, we identify the answer yes with the number 0 and the answer no with the number 1. We now describe the objects computed by such algorithms.

A k–ary relation is a subset of ω^k. We use capital Latin letters (generally P, Q, and R) for relations. If R is a relation, we usually write $R(\vec{x})$ for $\vec{x} \in R$. If R is 2–ary, we may also write $x\ R\ y$ for $R(x,y)$.

A 1–ary relation is simply a set of numbers. We understand set to mean set of numbers; we will use the word class for other kinds of sets. We use A and B for sets.

If R is a k–ary relation, the representing function of R, designated by χ_R, is the k–ary total function defined by

$$\begin{aligned}\chi_R(\vec{x}) &= 0 \qquad \text{if } R(\vec{x}),\\ &= 1 \qquad \text{otherwise.}\end{aligned}$$

A relation R is computable if the function χ_R is computable. We adopt the convention that whenever we attribute to a relation some property usually attributed to a function, we are actually attributing that property to the representing function of the relation.

3. The Basic Machine

To define our class of functions, we introduce a computing machine called the basic machine. It is an idealized machine in that it has infinitely much memory and never makes a mistake. Except for these features, it is about as simple as a computing machine can be.

For each number i, the computing machine has a register $\mathfrak{R}i$. At each moment, $\mathfrak{R}i$ contains a number; this number (which has nothing to do with the number i) may change as the computation proceeds.

The machine also has a program holder. During a computation, the program holder contains a program, which is a finite sequence of instructions. If N is the number of instructions in the program, the instructions are numbered 0, 1, ..., N–1 (in the order in which they appear in the program). The machine also has a counter, which at each moment contains a number.

To use the machine, we insert a program in the program holder; put any desired numbers in the registers; and start the machine. This causes 0 to be inserted in the counter. The machine then begins executing instructions. At