Chapter 3

Random number

For a given finite $\{0,1\}$-sequence x, let us consider a computer program which produces it. If x can be produced by a short program, it is considered to be regular, and otherwise, it is considered to be irregular. Consequently, it is a good idea to call x a random number if a very long program is needed to produce it ([5, 18, 19, 20, 26]). To make this idea precise and universal, we use the notion of partial recursive function ([5, 18, 19, 20, 21, 26], cf. [23, 54]). In this chapter, two fundamental theorems (Theorem 3.3 and Theorem 3.6, whose detailed proofs are not given here) are introduced as the basis of discussion, from which other theorems are derived. In particular, Theorem 3.15, which asserts that it is impossible to judge whether a given $\{0,1\}$-sequence is a random number or not, and Theorem 3.20, which says about the relation between random numbers and statistical tests, are very important.

The notion of random number may not directly solve practical problems, but recognizing it brings us profound understanding of the Monte Carlo method as well as probability theory itself (§ 3.6).

3.1 Partial recursive function

Modern computers can deal with many kinds of data; as input, data from keyboard, mouse, scanner, and video camera, . . . , as output, document, picture, sound, video, control sequence for electronic machine, ... But in the final analysis, all of them are binary strings, i.e., finite $\{0,1\}$-sequences.[†1] Since each finite $\{0,1\}$-sequence can be associated with a non-negative integer via dyadic expansion (§ 3.1.3), all data that computers deal with can be essentially regarded as non-negative integers. Namely, any action of computer can be regarded as a function $f : \mathbb{N} \to \mathbb{N}$.

Each action of computer is determined by a program, which, just like all input/output data, can be regarded as a finite $\{0,1\}$-sequence, or a non-negative integer, too. The set of all programs is therefore a countable set. In other words, among uncountably many functions $f : \mathbb{N} \to \mathbb{N}$, only countably many ones can be realized by actions of computer.

The notion of partial recursive function is used to express the actions of computer mathematically. By this notion, every action of computer, including infinite loops that do

[†1] Here we do not assume data processing of infinite input or infinite output.