

# CHAPTER 11

## 数列・漸化式と素数のふるまい

---

本章では数列と漸化式の扱いを学んだあと、整数に関するいくつかの予想（コラッツの予想，双子素数，素数分布）を数値実験してみよう。

また本章以降は，*Mathematica* の「プログラミング言語」的な側面にも光をあてる．具体的には **If** を用いた条件分岐，**Nest** を用いた関数の反復など，応用面では欠かせない重要な手法を学んでいく．

---

### 11.1 コラッツの予想

整数論で有名な予想といえば，90年代中ごろに解決された「フェルマー予想」や，未解決のものでは「ゴールドバッハ予想」，最近話題になった「ABC予想」などがある．また，素数分布に関する「リーマン予想」は，解析的数論とよばれる分野に属する未解決問題である．

さてみなさん，次の予想をご存知だろうか？

#### コラッツの予想 (the Collatz conjecture)

与えられた自然数  $n$  に対し，偶数ならば 2 で割る，奇数ならば  $3n + 1$  にする，という操作を繰り返すと，必ず

$$\dots \mapsto 1 \mapsto 4 \mapsto 2 \mapsto 1$$

というサイクルに落ち着く．

現象論的で地味な雰囲気漂う予想だが，角谷の問題， $3n + 1$  予想，シラキューズ予想などさまざまな呼び名で知られる，それなりに有名な未解決問題なのである．

試しに  $n = 7$  としてやってみよう。手続き通りにやっていると、

$$7 \mapsto 22 \mapsto 11 \mapsto 34 \mapsto 17 \mapsto 52 \mapsto 26 \mapsto 13 \mapsto 40 \mapsto 20 \\ \mapsto 10 \mapsto 5 \mapsto 16 \mapsto 8 \mapsto 4 \mapsto 2 \mapsto 1$$

となり、ちゃんと予想通りの結果になる。こういう一定の手続きを間違いなく、繰り返し実行するには、コンピューターの助けを借りるのが良い。もし *Mathematica* に上の数列を生成させたかったら、どのような機能が必要だろうか？少なくとも、次の3つが挙げられるだろう：

- ア. 与えられた数  $n$  が奇数が偶数かを判定する。
- イ. その結果に応じ、 $3n + 1$  か  $2n$  を出力する関数を定義する。
- ウ. その関数を出力が 1 になるまで繰り返す。

*Mathematica* にはこれらを実現する組込み関数がちゃんと実装されている。その詳細に入るまえに、ちょっと寄り道して *Mathematica* における数列と漸化式の扱いについて概説しておこう。

## 11.2 漸化式による数列の定義

数列は自然数  $n$  に数  $a_n$  を対応付ける関数の一種だとも考えられる。*Mathematica* では数列をリストとして扱うのが簡単でよいが、関数  $n \mapsto a(n)$  として数列を定義することもできる。<sup>\*1</sup>ここでは「与えられた漸化式を解く」というテーマをとおして、その方法を学んでいこう。

まずは典型例として、**フィボナッチ数列** (Fibonacci's sequence) がみたく漸化式

$$F_1 = 1, F_2 = 1, F_{n+2} = F_{n+1} + F_n$$

を考えよう。*Mathematica* でこの漸化式を考えるときは、「関数  $\text{fib} : n \mapsto F_n$  の値を  $n = 1$  のときから順に、帰納的に定義する」という方法をとる。<sup>\*2</sup>

[1] 漸化式でフィボナッチ数列を定義する：

```
fib[1] := fib[1] = 1; fib[2] = 1;
fib[n_] := (fib[n] = fib[n - 1] + fib[n - 2]);
```

<sup>\*1</sup> 他のプログラミング言語における「配列」に近い考え方だが、あらかじめ配列の長さを指定しない点、値の型を指定しない点が異なる。

<sup>\*2</sup> フィボナッチ数列はすでに組込み関数になっていて、 $F_n$  は `Fibonacci[n]` で与えられる。またここで紹介する手法以外にも、組込み関数 `RecurrenceTable` を用いる方法もある。くわしくはドキュメントセンターを参照せよ。