

数列は、「数の並びの規則性に着目しよう」という話題として語られる。  
数の並びの規則性から、賢く足し算をする知恵も導かれる。

### 0.1.1 数列を語る言葉

ある規則によって数を並べた列を**数列**という。

$$a_1, a_2, a_3, \dots, a_n, \dots$$

無限個の数が並んでいるものは**無限数列**、有限個の数が並んでいるものは**有限数列**という。

並んでいる数を **項**といい、最初数は**初項**、有限数列の場合は最後の数を**末項**という。  
先頭から数えて  $n$  番目の数を第  $n$  項、これは**一般項**ともよばれる。

具体的な数をそのまま1つずつ並べて書いてもよいが、長くなってしまう上、規則性が見えにくい。

一般項だけを  $\{\}$  で囲むことで、一般項で表される数の集合として

$$\{a_n\}$$

と書くことも多い。

\* \* \*

さて、重要な規則性を持つ数列には、名前がつけられている。  
まずは簡単なものから見ていこう。

### 0.1.2 等差数列

次のような規則で定まる数列は**等差数列**とよばれる。

初項  $a_1$  に次々と一定の数  $d$  を足してつくられている数列



等差数列では、隣り合った2つの項の差がいつも一定の値  $d$  になり、この  $d$  を等差数列の **公差** という。

### 0.1.3 漸化式

初項と、項と項の関係を表す式があれば、その式によって初項の次の数、またその次の数、… というように、数列を復元することができる。

これはいわばドミノ倒しのようなもので、具体的に触れるのは初項だけでよい。あとは規則に従って数が決まっていく。

### 0.1.4 「離散的な関数」としての視点

数を「並べる」ということは、位置を表す自然数と、その位置にある数とのマッピング（対応づけ）としても捉えられる。

すなわち、位置を与えたらその位置にある数を返す関数

$$f(n) = a_n$$

という形で数列を捉えることもできる。

このとき、数列という名前の関数  $f$  は、 $n$  から  $a_n$  を定める規則のことになる。

自然数  $1, 2, 3, \dots$  のそれぞれに対して、数  $a_1, a_2, a_3, \dots$  が定まっているとき、この対応づける規則を **数列** という。



[ Todo 1: 図：点で数列をプロットしたグラフ ]

$f(n)$  のグラフを書こうとしても、「線」にはならない。

$n$  は自然数 ( $1, 2, 3, \dots$ ) という飛び飛びに並んだ数であるため、各自然数の間の数に対しては  $f(n)$  の値が決まらない。そのため、 $f(n)$  はぽつりぽつりと点を並べて表すしかない。

このように、飛び飛びに並んでいるものは **離散的** とされる。

離散的なデータを扱う場面では、データを数列（離散的な関数）として見て調べる視点も重要になる。



Zebra Notes

Type	Number
todo	1