

# 第 1 章

## フィボナッチ数と行列

77 回生 Sugsugar

### 1.1 はじめに

はじめまして、Sugsugar です。中三です。普段は競技プログラミングをしています。あと ZONE を爆発させるのが得意です。

今回はフィボナッチ数に見る行列累乗の話について書いていこうと思います。なお、計算量に関してランダウの  $O$  記法というものを今回は使うので、気になった人は調べてみてください。

### 1.2 行列とは？

行列と聞いて queue を思い浮かべる人もいるでしょうが、今回は数学の行列です。具体的には、以下のようなものです。

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

たぶん高校数学でやると思います。これを使ってフィボナッチ数を高速に求めよう、というのが今回の話です。

### 1.3 行列累乗とは？

同じ行列を  $k$  回かけ算する (行列積の計算をする) ことを行列の累乗と呼び、

$$A^k = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdots \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

このように表します。なお、行列積については [こ↑こ↓](#) では次に、この計算を高速化する方法について書こうと思います。

### 1.4 繰り返し二乗法

単純に行列積を  $k$  回行うのは (行列のサイズを無視したとして)  $O(k)$  回かかってしまいます。さらに高速化して  $O(\log k)$  で求めることを考えてみます。

まず、行列ではなく素直に正整数  $a, n$  について  $a^n$  を求めることを考えます (なお、実際は  $n$  が大きくなるにつれ  $a \neq 1$  のとき  $a^n$  の値もかなり大きくなってしまいますので実際は mod を取ることを考えます)。そして、この計算は  $O(\log n)$  で求めることができます。その方法について以下に示していきます ( $a = 3, n = 22$  とします)。

まず、指数法則より  $3^22 = (3^11)^2$  となります。そして、 $3^11$  を指数的に分解していきたいのですが、半分に割ることはできません (かなしい)。ではどうするかというと単純で、 $3^11 = 3^1 * 3^10$  とすれば片方の指数を偶数にすることができました。

同様にして、 $3^{2^2} = (3^1 * (3^1 * ((3^1)^2)^2)^2$  と表すことができます。整理すると、 $3^{2^2} = 3^2 * 3^4 * 3^{16}$  と表すことができます。これはつまり、指数を2のべき乗の数で分解したということです。さて、ではこの分解した2のべき乗(ここで  $2^k$  とします)の指数の累乗についてはどのように求めればよいのでしょうか? そう、例えば  $3^2 = (3^1)^2$  ですし、 $3^4 = (3^2)^2$  です。つまり、 $3^{(2^k)}$  は3を  $\log_2 k$  回2乗した数と考えることができます。このことから  $O(\log k)$  で  $a^n$  の値を求めることができます。これを繰り返し二乗法といいます。コードは [こ↑こ↓](#) に書いてありますので是非見てみてください。決して言葉で説明するのがめんどくさくなったというわけではないですよ。

## 1.5 行列で応用

整数ができるのならば行列でもできそうなことは直感的にわかりますが(本当? )、実際にできます。なので行列累乗の計算は行列のサイズを無視すると  $O(\log k)$  で求められます。いい話。

## 1.6 フィボナッチ数で応用

さて、長々と話してきましたが本題です。忘れていた人にもう一度言うと、フィボナッチ数をこの行列累乗で求めたいという話です。フィボナッチ数は以下のように表せました。

$$f_0 = 0, f_1 = 1, f_n = f_{n-1} + f_{n-2} (n \geq 2)$$

では、このような行列  $A$  が存在すると嬉しいと思いませんか?

$$\begin{bmatrix} f_{n+1} \\ f_n \end{bmatrix} = A \begin{bmatrix} f_n \\ f_{n-1} \end{bmatrix}$$

ここで行列  $A$  について考えましょう。ヒントとして、 $A$  は  $2 \times 2$  の行列です。そして、 $f_{n+1} = f_n + f_{n-1}$ ,  $f_n = f_n$  であり、行列積の計算方法を考えると因数分解みたいなイメージで考えることができます。

さて、では答えです。

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

答えはこうなります。実際に計算してみるとよりわかるかもしれません。

このことから、さらに以下のこともわかります。

$$\begin{bmatrix} f_n \\ f_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_{n-2} \end{bmatrix}$$

まあ当然といえば当然です。このことから、

$$\begin{bmatrix} f_n \\ f_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_{n-2} \end{bmatrix}$$

となることもわかります。さて、これを繰り返すことにより

$$\begin{bmatrix} f_{n+1} \\ f_n \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} f_1 \\ f_0 \end{bmatrix}$$

つまり、

$$\begin{bmatrix} f_{n+1} \\ f_n \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

となることがわかりました！あとはウイニングランです。

この累乗されているところはどうか求めますでしょうか？もちろん愚直にやると遅いです（というかわざわざ行列で表した意味がない）。そう、行列累乗です。これで計算量が  $O(k) \rightarrow O(\log k)$  まで落とすことができました。うれしいですね。なお、行列積の計算量は  $n \times n$  行列のときに  $O(n^3)$  かかりますが、今回  $n = 2$  なので無視してもよいです。というか  $O(\log k)$  すら定数と言い張る人がいるくらいなので気にしたら負けです。

## 1.7 フィボナッチ数の拡張

さて、今回は

$$f_0 = 0, f_1 = 1, f_n = f_{n-1} + f_{n-2} (n \geq 2)$$

としていましたが、

$$f_0 = f_1 = 0, f_2 = 1, f_n = f_{n-1} + f_{n-2} + f_{n-3} (n \geq 3)$$

のようにトリボナッチ数列について考えてみようとするのは自然な考えです。これは

$$\begin{bmatrix} f_{n+2} \\ f_{n+1} \\ f_n \end{bmatrix} = A \begin{bmatrix} f_{n+1} \\ f_n \\ f_{n-1} \end{bmatrix}$$

を満たす  $3 \times 3$  行列  $A$  について考えれば先ほどと同じようにできます。これをさらに拡張していくとこの行列  $A$  に関してある規則が見えてくるはずですが、ここでは答えを言わないので、皆さん自身で考えてみてください。

## 1.8 おわりに

フィボナッチ数をコンピュータで高速に求めたいという話でした。僕は競技プログラミングというものを普段していて、この行列累乗というアルゴリズムは AtCoder でいう水～黄色で学ぶアルゴリズムとなっています。また、繰り返し二乗法は AtCoder でいう茶～緑色で学ぶアルゴリズムです。一見ただただ足すだけだと思っていた数列を行列を用いることによって計算量が大幅に激減するのはアルゴリズムの醍醐味であると思います（実際、 $10^{18}$  項目だと愚直にやると人間では考えられないほど時間がかかりますが、一方でこの方法だと 1 秒もかからずにできます）。アルゴリズムにあまり詳しくなくてもフィボナッチ数というよく知られている話によって少しでも楽しさを理解していただけたら幸いです。

P.S. そういえば、サンプル版ではビネの公式について話す旨について書いていましたが、これは先ほどの  $2 \times 2$  の行列  $A$  を用いることでなんと導出することができます。いつか必ず書きます（もしかしたら文化祭のポスターとして展示しているかもしれません）。楽しみにしててください。