

1.2 フィボナッチ数列の2項間の比

フィボナッチ数列には面白い性質がある。そのひとつは、隣り合う2項の比を調べることで見えてくる。1, 1 から始まるフィボナッチ数列を2項ずつペアにし、

$$\frac{1}{1}, \frac{2}{1}, \frac{3}{2}, \frac{5}{3}, \frac{8}{5}, \frac{13}{8}, \frac{21}{13}, \frac{34}{21}, \dots$$

のように分数を作る。この程度は暗算や電卓で計算できるが、蝶道を這い回っているのだから $\text{T}_\text{E}_\text{X}$ で試したい。

```
\newcommand\ffailure{\par
  \newdimen\fa \newdimen\fb \newcount\fc \newdimen\ft
  \fa=1pt \fb=1pt
  \loop
    \advance\fa\fb
    \ft=\fa \fa=\fb \fb=\ft
    \the\ft/\the\fa = \divide\ft\fa \the\ft\par
    \advance\fc1
  \ifnum\fc<5 \repeat
\noindent}
```

しかし、『フィボナッチ数列の2項間の比は\ffailure である。』と書くと『フィボナッチ数列の2項間の比は

2.0pt/1.0pt= 0.00003pt

3.0pt/2.0pt= 0.00002pt

5.0pt/3.0pt= 0.00002pt

8.0pt/5.0pt= 0.00002pt

13.0pt/8.0pt= 0.00002pt

である。』という失敗例が出力される。マクロに含まれている\par と\noindent は、このマクロが失敗することは承知の上で、出力を整えるために入れたものである。もちろん、\par や\noindent は $\text{T}_\text{E}_\text{X}$ を真つ当に使う際に書く命令で、マクロのための命令ではない。

さて、何がまずかったのだろう。比を計算すると小数值が出るので、\newcount で宣言する整数変数では都合が悪い。そこで**寸法**に使われるレジスタ用の命令である\newdimen を用いている。 $\text{T}_\text{E}_\text{X}$ では寸法には単位 “pt” が使われるので、変数への代入も pt 付きの値でなければならない。そうしておけば、\divide\ft\fa の計算は小数值が期待されるだろう。結果は pt がくっついているものの、割り算式には正しい式が与えられていることは間違いない。ちなみに、マクロは割り算を (第3項)/(第2項) から始めていることに注意されたい。しかし、右辺の値は小数值ではあるが、期待された値ではない。 $\text{T}_\text{E}_\text{X}$ には独特の癖があるのだ。

問題を解決する前に、マクロについて補足しておこう。基本的には前のものと大差ないが、比の値を $A/B = R$ の形で示したかったので `\the\t/\the\a = \divide\t\a \the\t\par` という書き方をした。ここで実際に出力される部分は `\the\t/\the\a = \the\t` であり、`\divide\t\a` と `\par` は処理は行いが文字として出力されるものではない (`\par` は“改行”という形で出力されるが)。

`\the\t` は `\number\t` と同様、`\t` の現在値を出力するが、`\number` が 10 進数に展開するのに対し `\the` はトークンの並びに展開する。トークンの並びというのは 5.0pt plus 2.0fil という展開もありということだ。早い話、単位や計算式まで出力される。実際、`\the` を `\number` に変えて出力すると、1 行目に出力された 2 項間の比

$$2.0\text{pt}/1.0\text{pt} = 0.00003\text{pt} \quad \text{は} \quad 131072/65536 = 2$$

と出力される。計算式としては正しくても、期待した結果にはなっていない。

そこで、小数値を求める割り算のために、専用のマクロが必要になる。ここでは新しく書くことはせず、`tmtmath.sty` ファイルで使っている除算マクロ `\dimendiv` を使っておこう。

```
\makeatletter
\newcommand\fsuccess{\par
  \newdimen\a \newdimen\b \newcount\c \newdimen\t
  \a=1pt \b=1pt
  \loop
    \advance\a\b
    \t=\a \a=\b \b=\t
    \the\b/\the\a = \dimendiv(\strip@pt\b, \strip@pt\a)\the\dimen@\par
    \advance\c1
  \ifnum\c<15 \repeat
\noindent}\makeatother
```

これで、『フィボナッチ数列の 2 項間の比は `\fsuccess` である。』と書くと『フィボナッチ数列の 2 項間の比は

$$2.0\text{pt}/1.0\text{pt} = 2.0\text{pt}$$

$$3.0\text{pt}/2.0\text{pt} = 1.50003\text{pt}$$

$$5.0\text{pt}/3.0\text{pt} = 1.66678\text{pt}$$

$$8.0\text{pt}/5.0\text{pt} = 1.60004\text{pt}$$

$$13.0\text{pt}/8.0\text{pt} = 1.62506\text{pt}$$

$$21.0\text{pt}/13.0\text{pt} = 1.61551\text{pt}$$

$$34.0\text{pt}/21.0\text{pt} = 1.61916\text{pt}$$

$$55.0\text{pt}/34.0\text{pt} = 1.61778\text{pt}$$

$89.0\text{pt}/55.0\text{pt} = 1.61832\text{pt}$
 $144.0\text{pt}/89.0\text{pt} = 1.61815\text{pt}$
 $233.0\text{pt}/144.0\text{pt} = 1.61816\text{pt}$
 $377.0\text{pt}/233.0\text{pt} = 1.61812\text{pt}$
 $610.0\text{pt}/377.0\text{pt} = 1.61813\text{pt}$
 $987.0\text{pt}/610.0\text{pt} = 1.61813\text{pt}$
 $1597.0\text{pt}/987.0\text{pt} = 1.61813\text{pt}$

である。』が出力される。やったね。でも、出力は相変わらず pt がくつついているし、マクロには変な記号が使われている。何をしているか見ていこう。

まず、“@” が気になるね。でも、`\strip@pt` という命令自体が “@” を含んでいる命令なのだ。ところが $\text{T}_{\text{E}}\text{X}$ は “@” を特別な記号として扱っているので、普通に使うと受け付けてくれない。そのため、“@” を使う前に `\makeatletter` を、使った後に `\makeatother` を宣言するのが決まりだ。

`\dimendiv` は `\dimendiv(21, 13)` のように使うことで $\frac{21}{13}$ を計算するのだが、`\b` と `\a` は寸法であるから `\dimendiv(\b, \a)` としたのでは `\dimendiv(21pt, 13pt)` になって具合が悪い。`\strip@pt` は pt を取り除く命令なのである。それなら最初から普通に整数を代入すればよいと思うかもしれないが、割り算は当然小数値どうしても行うことがあるので、このような仕様にしてあるのだ。で、`\dimendiv` は割ることしかやらないので、結果の出力は `\the\dimen@`で行っている。`\the\b` じゃないの？と思うかもしれないが、これも仕様である。済まんね。

ところで、フィボナッチ数列というのは、何も 1, 1 から始まると決められているわけではない。たとえば 10, 3 から始めてもよい。すると

10, 3, 13, 16, 29, 45, 74, 119, ...

のようになって、1, 1 から始まる数列とは違うものになる。しかし、2 項の比についてはそうでもない。

```

\makeatletter
\newcommand\fstartdiff{\par
  \newdimen\a \newdimen\b \newcount\c \newdimen\t
  \a=10pt \b=3pt \c=1
  \loop
    \advance\a\b
    \t=\a \a=\b \b=\t
    $\the\t/\the\a = \dimendiv(\strip@pt\t, \strip@pt\a)\the\dimen@\par
    \advance\c1
  \ifnum\c<9 \repeat
\noindent}\makeatother

```

これで、『10, 3 から始めると\fstartdiff になる。』と書けば『10, 3 から始めると

$$13.0pt/3.0pt = 4.33339pt$$

$$16.0pt/13.0pt = 1.23083pt$$

$$29.0pt/16.0pt = 1.81259pt$$

$$45.0pt/29.0pt = 1.55179pt$$

$$74.0pt/45.0pt = 1.64453pt$$

$$119.0pt/74.0pt = 1.6082pt$$

$$193.0pt/119.0pt = 1.62195pt$$

$$312.0pt/193.0pt = 1.61673pt$$

になる。』が出力される。前の出力と比べてほしいところがある。 pt が斜体になったところではない。等号の前後の空気が整っている方だ。マクロを注意深く見れば分かると思うが、今回の出力は数式モードで記述した。するとこうなるわけだ。こんなところにも $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の律儀さが見え隠れする。

さて、いろいろな入力を試してみれば、いつでも $1.6\cdots$ 程度の値に収まるように思える。1 にまつわる道を這っているので、目にする数値は当然 1 から始まる。この数値が何かは、もうしばらく這い回ったときにはっきりするだろう。