## 4.3 双子素数

素数の話題は尽きることがない。話題のひとつに**双子素数**というものがある。2 を除けば素数は奇数であるが、ときどき 17, 19 や 41, 43 のように連続して素数が現れることがある。これらを双子素数と呼んでいる。

素数が無数にあることは証明されている。それはユークリッド[1]による証明が有名である。証明の大筋は次のようなものだ。

素数が有限個しかないと**仮定**し、素数を $2, 3, 5, \ldots, P$ とする。このとき $2\cdot3\cdot5\cdots P+1$ なる数を作れば、この数は有限個のすべての素数で割ると 1 余ることになる。つまり、$P$ より大きい新たな素数となるから、素数が $P$ までしかないことに矛盾してしまう。ここで矛盾を生じさせないためには $2\cdot3\cdot5\cdots P+1$ が合成数であると考えなくてはならない。合成数は何らかの素数で割れなければならないのだが、有限個の素数で割れなかったので、これら以外の素数で割れなくてはならない。つまり、この場合も新たな素数が存在することになって、素数が $P$ までしかないことに矛盾してしまう。これらの矛盾は素数が有限個であると仮定したことに起因している。すなわち素数は有限個ではない。

このように、ある仮定から始めて矛盾を引き出し、結果、仮定が間違っていると結論する証明方法は**背理法**と呼ばれる。以前、濃度の話でも同様の説明をしたことを覚えているだろうか。あのときも背理法を使って説明していたのである。

ところで、素数がどれぐらいの頻度で現れるかとか、双子素数も無数にあるのかなど、素数には未解決の問題が多い。素数の頻度というのも変な感じであるが、ガウス[2]は素数が現れる頻度について、次のような見解を持っていた。

$$1 \text{ から } N \text{ までには、素数はおよそ} \frac{N}{\log_e N} \text{個含まれる}$$

たとえば $N = 100$ とすると、

$$\frac{100}{\log_e 100} \approx 21.7$$

だが、実際は 25 個の素数がある。ぴったりではないが、まあ近いところをついている。式は $N$ が大きいほど、より正しい頻度を教えてくれる。もし $N$ を与えたとき、本当にぴたり正確な素数の数を求める式が発見されれば、画期的なことであるが、いまのところそのような式は発見されていない。またガウスの式とは別の、素数の頻度を計算する式はあるが、ガウスの式の美しさにはかなわないのだ。何とも不思議なことである。

[1] ユークリッド（330?B.C.–275?B.C.）：ギリシアの数学者。
[2] カール・フリードリヒ・ガウス（1777–1855）：ドイツの数学者。

双子があれば三つ子も考えたくなるのが人情で、3, 5, 7 のような三つ子素数がどれくらいあるかと考えてしまう。だが結論を言おう。これ以外の三つ子素数はない。連続する 3 つの奇数は三つ子素数にはならないのだ。理由は単純だ。ある連続する 3 個の奇数を $2m+1$, $2m+3$, $2m+5$ とおいてみる。偶数と奇数は交互に現れるので、□を偶数、△を奇数とした数の並びは

$$\ldots,\quad \triangle,\quad \square,\quad 2m+1,\quad \square,\quad 2m+3,\quad \square,\quad 2m+5,\quad \square,\quad \triangle,\quad \ldots$$

(a) $\quad *\qquad\qquad *\qquad\qquad \underline{*}$
(b) $\quad\quad *\qquad\qquad \underline{*}\qquad\qquad *$
(c) $\quad\qquad \underline{*}\qquad\qquad *\qquad\qquad *$

となっているはずだ。ところで連続する数の並びには、3 個ごとに 3 の倍数（*）が存在している。ほらね。3 個ごとに数を選べば、(a), (b), (c) のどの場合でも $2m+1$, $2m+3$, $2m+5$ のどれかは 3 の倍数（$\underline{*}$）になってしまうね。だから $2m+1$, $2m+3$, $2m+5$ がすべて素数なんてことはあり得ないのだ。

三つ子素数がないと分かったところで、双子素数に集中しよう。と言っても場当たり的に双子素数を探すのでは効率が悪い。徹底した調査で、どのような条件のとき双子素数が現れるのかが分かれば、スクリプトも書きやすくなる。そこで、ここでは次の単純な事実をもとに、双子素数を見つけていこう。

2, 3 を除くすべての素数は $6m \pm 1$ の形をしている

そう、素数は決して $6m \pm 2$ や $6m \pm 3$ のような形をしていないのだ。誤解しないでもらいたいことがある。私は「素数は $6m \pm 1$ の形をしている」と主張しているのであって、決して「$6m \pm 1$ の形をしている数は素数である」と言っているのではない。**逆**は必ずしも正しくないとは、まさにこのことを言う。そして、ここでも少し考えれば、すべての素数が $6m \pm 1$ の形をしている理由も分かるだろう。だって、$6m \pm 2 = 2(3m \pm 1)$、$6m \pm 3 = 3(2m \pm 1)$ なんだから素数になり得ないのだ。

あれ？　$6m \pm 4$ や $6m \pm 5$ は考えなくてもよいの、と思うかもしれないね。$6m \pm 4$ は上と同じ理由で素数になり得ないとしても、$6m \pm 5$ は素数になる可能性が残るんじゃないの。でも、そうじゃない。$6m \pm 4$ や $6m \pm 5$ は考える必要はないのだ。

数を $6m \pm 1$ の形で見直すことは、6 で割った余りが 1 か $-1$ である数と見ている。余りが $-1$ というのは違和感があるかもしれないが、6 で割った余りが 0, 1, 2, 3, 4, 5 ということを、6 で割った余りが $-2$, $-1$, 0, 1, 2 と見直しているだけである。具体的には、17 は 6 で割った余りは $17 = 6 \times 2 + 5$ より 5 であると見るか、$17 = 6 \times 3 + (-1)$ と見るかの違いだ。いずれにしても、余りは 5 種類しか出ないのだ。

このような見方は**剰余類**、つまり剰余に関するグループで分類することなのである。17 以外にも 11 や 23 は 6 で割ると 5 余るので、同じグループに属する。話を簡単にするため正の整数に限って言うと、6 で割った余りが 5 のグループ $R_5$ は

$$R_5 = \{5,\ 11,\ 17,\ 23,\ 29,\ 35,\ 41,\ \ldots,\ \ldots\}$$

である。同様にして

$$R_0 = \{6,\ 12,\ 18,\ 24,\ 30,\ 36,\ 42,\ \ldots,\ \ldots\}$$

であるから、整数は $R_0$, $R_1$, $R_2$, $R_3$, $R_4$, $R_5$ の 6 グループにきっちり分類できる。

では、さっき余りが $-1$ という見方をしたグループは何だろう。実は、それは $R_5$ である。$R_5$ に属する数はすべて $6m + (-1)$ の形にできるからだ。なぜなら $m$ をひとつずらして $m = m' - 1$ としてみれば、

$$R_5 = 6m + 5 = 6(m' - 1) + 5 = 6m' + (-1)$$

となることが確認できる。したがって $R_5 = R_{-1}$ がいえる。おっと、庭いじりをしているつもりが、別の方向に向かってしまった。庭いじりに戻ろう。

さて、そういうことなら話は簡単だ。$6m - 1$ が素数であるかどうか調べ、素数であるときに限り $6m + 1$ が素数であるかどうか調べればよい。運良く 2 つとも素数なら、それが双子素数ということだ。

(ghci env.)

```
*Main> [(6*n-1, 6*n+1) | n <- [1..100], isprime (6*n-1) && isprime (6*n+1)]
[(5,7),(11,13),(17,19),(29,31),(41,43),(59,61),(71,73),(101,103),(107,109),
(137,139),(149,151),(179,181),(191,193),(197,199),(227,229),(239,241),(269,
271),(281,283),(311,313),(347,349),(419,421),(431,433),(461,463),(521,523),
(569,571),(599,601)]
```

`isprime` 関数があるので、対話モードで 1 行のスクリプトで済んでしまった。この場合は、`6*n-1` と `6*n+1` の 2 数が素数になっていれば、それらをペアにしてリスト入りさせるものである。`n` に 1 から 100 まで順に代入し、`isprime` 関数で `6*n-1` と `6*n+1` が共に素数になるものだけが `(6*n-1, 6*n+1)` の組になるのである。

`isprime` 関数が、`True` か `False` を返す関数であったことを思い出してほしい。そんなわけで `isprime (6*n-1)` と `isprime (6*n+1)` が共に真になるかどうかを調べるために、`&&`関数を使っている。`A && B` は、`A` と `B` が共に真であるときに限り `True` を返す。したがって、`n <- [1..100]` のうち、条件を満たすペアが表示されるのである。ただし、$6m \pm 1$ 型の奇数だけを調べたので、いちばん最初に現れるべき双子素数 (3, 5) が含まれていないのはご愛嬌と思ってほしい。

しかし、それ以上に不自然なことは、100 までの $n$ を調べると 600 前後の双子素数を求めてしまう点だ。$n$ に 100 を与えたら、100 までにどれぐらいの双子素数があるかを表示する方が自然だろう。そうなると、別のスクリプトを書かなければならない。少しぐらい多めに双子素数を表示したからといって困ることはないので、1 行のスクリプトで目的を達したことを素直に喜ぼう。