8.4 ガウスの絨毯

では、ガウス素数の一覧でも見ることにしよう。整数 n が、ガウス素数かどうか調べるスクリプトは簡単に対話モードの中で定義したが、一覧を見ようとするならコードを書いた方がよい。

[hs script]
```
cnumfactor :: Int -> (Int, Int)
cnumfactor n
    | xs == []  = (n, 0)
    | otherwise = head xs
    where xs = [(a, b) | b <- [1..n], a <- [1..b], a^2 + b^2 == n]
```

前回の定義と同じなので、`[1..n]` の範囲指定は手を抜いたままだ。しかし、共役複素数の積に分解できない整数でもエラーにはならないので、そこでは手を抜いていない。

(ghci env.)
```
*Main> cnumfactor 13
(2,3)
*Main> cnumfactor 19
(19,0)
```

さあ、あとは虚部に虚数単位 i を付けて実部とくっつければよい。これは対話モードで定義してしまおう。

(ghci env.)
```
*Main> let gaussprime n = show (fst (cnumfactor n)) ++ " :+ " ++ show (snd
 (cnumfactor n)) ++ "i"
*Main> gaussprime 13
"2 :+ 3i"
```

ふむ、まずまずの結果かな。で、これをリストにして `mapM` へ渡せば一覧が出来上がるだろう。実数だけの世界で素数だったものが、複素数の世界でも共役複素数の積に分解できるかを調べるので、奇数だけを調べれば十分だ。本当は、素数だけを調べるべきだけどね。

(ghci env.)
```
*Main> mapM_ putStrLn [gaussprime n | n <- [1, 3..20]]
1 :+ 0i
3 :+ 0i
1 :+ 2i
7 :+ 0i
9 :+ 0i
11 :+ 0i
2 :+ 3i
15 :+ 0i
1 :+ 4i
19 :+ 0i
```

共役複素数の積にならない整数は、常に 0i が表示されてみっともないが、とりあえず目的は果たしている。ただ、どの行がいくつの整数の分解かは見づらい。そこで、大げさになるが

(ghci env.)

```
*Main> let gaussprime’ n = if (snd (cnumfactor n)) == 0 then show n else
show n ++ " = (" ++ show (fst (cnumfactor n)) ++ " :+ " ++ show (snd (
cnumfactor n)) ++ "i)(" ++ show (fst (cnumfactor n)) ++ " :- " ++ show
 (snd (cnumfactor n)) ++ "i)"
```

としてみよう。ワオ！　正気か。ここでしたことは、単に show をつなげてそれなりの表示ができるように定義しただけである。すると、こうなる。

(ghci env.)

```
*Main> mapM_ putStrLn [gaussprime’ n | n <- [1, 3..20]]
1
3
5 = (1 :+ 2i)(1 :- 2i)
7
9
11
13 = (2 :+ 3i)(2 :- 3i)
15
17 = (1 :+ 4i)(1 :- 4i)
19
```

うーむ。価値あることなんだろうか。まあ、こうして見ると、複素数の世界では 5、13、17 は素数と呼べず、3、7、11、19 は相変わらず素数ということが分かる。9 や 15 はもともと合成数だ。相変わらずの素数と合成数の区別は自分の目でするしかない。そうすると、この区別も **Haskell** にやってもらいたいけれど、ちょっと気が進まないね。

気が進まない理由はスクリプトを書くのが大変だからではない。整数に限って調べても十分とは言えないからだ。ここで調べた数は、あくまでも整数である。つまり、ガウス平面の実数軸上の整数だけを調べている。でも、実部、虚部ともに整数である複素数は、ガウス平面いっぱいに広がっているのだ。

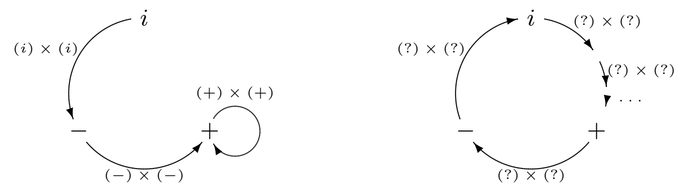
たとえば $11 + 10i$ は $(2 + 3i)(4 - i)$ に分解できる。そういう点で $11 + 10i$ は複素数の世界の合成数になっている。でも $4 + i$ は整数値を用いて複素数の積に分解できない。こういう複素数がガウス素数である。さあ、そうなるとちょっとやそっとじゃガウス素数を見つけることはできそうにない。それに、ガウス素数を見つけるスクリプトを書いたとしても、ガウス素数の分布は見えてこないだろう。もしガウス素数の分布を見ようと思えば、ガウス平面にガウス素数をプロットするのがよい。一目瞭然となるからだ。

でも、残念ながらジョウロとスコップではそれは無理だ。ガウス平面に広がるガウス素数の分布

は、実軸、虚軸、それに対角軸に対して対称な模様を作る。なかなかきれいな模様なので**ガウスの絨毯（じゅうたん）**などと呼ばれている。興味があればインターネットで調べてみるとよいだろう。

さらに、ガウス素数の亜種とでも言おうか、アイゼンシュタイン[1]素数というものもある。これは $a+bi$（a、b は整数）の形であるガウス素数の代わりに、$a+b\omega$（a、b は整数）の形である整数を定義することが出発点となる。ω は 1 の 3 乗根のひとつで、具体的には $\dfrac{-1+\sqrt{3}\,i}{2}$ または $\dfrac{-1-\sqrt{3}\,i}{2}$ のことである。どっちを ω と決めても、もう一方は ω^2 になるという性質を持っている。この性質のためにアイゼンシュタイン整数は、ガウス整数が ± 1 と $\pm i$ の 4 数を**単数**とするように、± 1、$\pm\omega$、$\pm\omega^2$ の 6 個の単数を持つことになる。すると、ガウス素数が 8 本の軸に対称な模様に分布するのと同様、アイゼンシュタイン素数は 12 本の軸に対称な模様に分布するのを目にすることができるのだ。なかなかに複素数の世界は興味深い。

ガウス素数やアイゼンシュタイン素数がきれいに整った様子を見ると、複素数の対称性にも関心がいく。前に $(i)^2 \to (-)$、$(-)^2 \to (+)$ から $(+)^2 \to (i)$ になれば、符号の輪が閉じて面白いと言ったのを覚えているだろうか。でも、残念ながら $(+)^2 \to (+)$ だから輪は閉じない。



けれども前々節で、平方ではなく平方根で考えると、$(-)$ の平方根 $\to$ (i)、(i) の平方根 $\to$ $(\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}i)$、$(\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}i)$ の平方根 $\to \cdots$ の行き着く先が $(+1)$ であると言ったはずだ。このことと $(+)$ の平方根のひとつが $(-)$ であることを合わせると、なんと輪が閉じるじゃないか。やっぱり複素数って不思議だ。

でも、庭いじりとしてはここまでだろう。8 週にわたって庭の隅をいじってきたが、その間に季節も変わり始めてしまった。これまでに土を掘り返したり種を蒔いたりしたけれど、しばらくそっとしておこう。違う季節には別の庭いじりの仕方があるはずだ。この先、庭に華やかさを加えるためにも、ぜひ君たち自身で数学や **Haskell** に親しんでみよう。絶対に新たな発見があるはずだから。

[1]フェルディナント・ゴットホルト・マックス・アイゼンシュタイン（1823–1852）: ドイツの数学者。ガウスに師事した。