8.3 ガウス素数

複素数の登場で世界が変わってしまった。とくに変わってしまうのが素数の世界である。なぜなら、いままでは 2 は素数の扱いをしていたが $2 = (1+i)(1-i)$ に分解できるからだ。しかし 3 はそうならない。3 は複素数の範囲にまで広げても素数であり続ける。

実数の世界で素数であったものが、複素数の世界で合成数になってしまう数にはどんなものがあるだろうか。そのようになる実数 x は

$$x = (a+bi)(a-bi)$$

を満たしているはずである。つまり、共役複素数の積でないといけない。**Haskell** は簡単にそういうものを見つけることができる。

(ghci env.)

```
Prelude> let gprime n = [(a, b) | a <- [1..n], b <- [1..n], a^2 + b^2 == n]
```

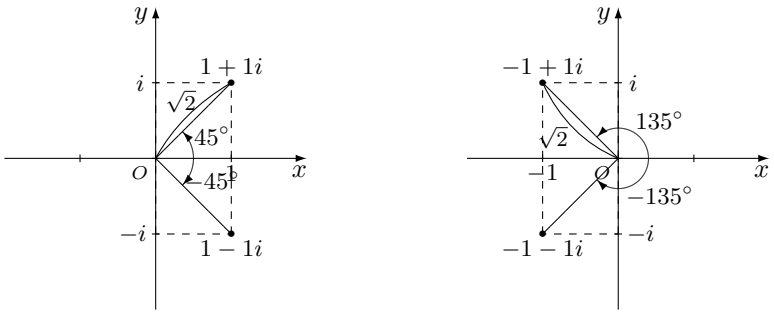
この程度なら対話モードで十分定義可能で、単に $a^2 + b^2 = n$ となるペアを探しているだけだ。

(ghci env.)

```
Prelude> gprime 2
[(1,1)]
Prelude> gprime 3
[]
```

2 と 3 を調べてみた結果、$2 = (1+1i)(1-1i)$ であること、また 3 は共役複素数の積に分解できないことが分かる。本当は `a`、`b` の範囲は `[1..sqrt n]` で定義するのが効率的だが、こうすると **Haskell** の型推論が働いて `[(1.0,1.0)]` と出力されてしまい煩わしい。しかし、これでは味気ないことも確かである。もう少し見栄えがよくならないだろうか。

見栄えの前に注意することがある。いま $2 = (1+1i)(1-1i)$ を見たのだが、本当は $a^2 + b^2 = 2$ を満たす $(a,\ b)$ の組は $(1,\ 1)$、$(1, -1)$、$(-1,\ 1)$、$(-1,\ -1)$ の 4 組がある。ただ、$(a+bi)(a-bi)$ に当てはめると、実際は $(1+1i)(1-1i)$、$(-1+1i)(-1-1i)$ の 2 組にまとめられる。



複素数を表すとき、実軸 x と虚軸 y の目盛りを組にして表す方法以外に、原点からの距離 r と実軸の正の方向となす角 θ で表す方法がある。これでも同じことであるのは図を見れば分かるだろう。たとえば $1+1i$ は、原点からの距離が $\sqrt{2}$ で、なす角は 45° である。三角比を用いれば

$$1+1i = \sqrt{2}\left(\frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i\right) = \sqrt{2}(\cos 45^\circ + i\sin 45^\circ)$$

である。同様にして $1-1i = \sqrt{2}(\cos 45^\circ - i\sin 45^\circ)$ であるが、$1-1i$ は -45° の角を持っているので $1-1i = \sqrt{2}\{\cos(-45^\circ) + i\sin(-45^\circ)\}$ と見た方がよい。これらの積は本当に 2 になるだろうか。計算してみると

$$\begin{aligned}
& \sqrt{2}(\cos 45^\circ + i\sin 45^\circ)\cdot\sqrt{2}\{\cos(-45^\circ) + i\sin(-45^\circ)\} \\
= \;& 2[\{\cos 45^\circ\cos(-45^\circ) - \sin 45^\circ\sin(-45^\circ)\} + i\{\cos 45^\circ\sin(-45^\circ) + \sin 45^\circ\cos(-45^\circ)\}] \\
= \;& 2[\cos\{45^\circ + (-45^\circ)\} + i\sin\{45^\circ + (-45^\circ)\}] \\
= \;& 2[\cos 0^\circ + i\sin 0^\circ] = 2
\end{aligned}$$

で、合っている。三角関数の加法定理を用いたので、詳しくない人は計算を信用してもらいたい。一般に、加法定理を用いると

$$r(\cos\theta + i\sin\theta)\cdot r\{\cos(-\theta) - i\sin(-\theta)\} = r^2(\cos 0^\circ + i\sin 0^\circ) = r^2$$

になることが示される。

$a+bi$ と $a-bi$ の組は実軸をはさんで上下対象の位置にあるから、実軸となす角はそれぞれ θ と $-\theta$ である。また、$r = \sqrt{a^2+b^2}$ であることに注意すると、2 数の積は a^2+b^2 であることが分かる。$(a+bi)(a-bi)$ に分解できる整数の条件が $a^2+b^2 = n$（n は整数）であったのは、以上のことが理由であった。

それなら、$a+bi$ と $a-bi$ の実軸となす角が $\pm 45^\circ$ でなく、たとえば $\pm 30^\circ$ でも $a+bi$ と $a-bi$ の積は r^2 だから、r^2 が整数になる組を見つければいいんじゃない？と思ったかもしれない。でも違う。なす角が $\pm 30^\circ$ では格子点上に複素数は存在しないのである。なぜなら、実軸となす角が 30° の場合、複素数は必ず $k(\sqrt{3},\ 1)$ の位置に現れる。つまり、複素数は $k\sqrt{3}+ki$ の形をしていることになる。このとき、$k\sqrt{3}$ と k は同時に整数値になることはない。

さあ、再び **Haskell** に働いてもらおう。しかし、すでに整数を共役複素数の積にするスクリプトは書いている。あとは表示の仕方だけが問題だったんだ。もう一度さっきのスクリプトで整数の分解を試みることにする。

(ghci env.)
```
Prelude> gprime 5
[(1,2),(2,1)]
```

5 が複素数の範囲で分解できるか調べたところだ。これより

$$5 = (1+2i)(1-2i) = (2+1i)(2-1i)$$

であることが分かるが、前に述べた通り、これ以外にも $\pm 1, \pm 2$ を組み合わせた分解もあるので、ガウス平面上ではこれらの点は 8 か所に現れる。しかし 8 点は、$1+2i$ を実軸、虚軸、そして、なす角 45° および 135° の直線に関して対象の位置にあるだけなのだ。要するに、5 の分解の本質的複素数は $1+2i$ だけと言っても過言ではない。それなら gprime の定義にもうひとつまみ条件を加えれば、本質的素数だけを出力する関数にできる。そのために a <= b を加えてみた。

(ghci env.)
```
Prelude> let gprime n = [(a, b) | a <- [1..n], b <- [1..n], a^2 + b^2 == n
, a <= b]
Prelude> gprime 5
[(1,2)]
```

これで本質的な共役複素数をただひとつ出力する関数になったので、こいつを利用して出力を工夫しよう。それには、こうする。

(ghci env.)
```
Prelude> show (fst (head (gprime 5))) ++ " :+ " ++ show (snd (head (gprime
 5))) ++ "i"
"1 :+ 2i"
```

ちょっと長ったらしいし、( ) が邪魔くさいけれど、show が今回の役目には適任である。長ったらしいことを隠すなら、gprime' を再定義しよう。

(ghci env.)
```
Prelude> let gprime' n = show (fst (head (gprime n))) ++ " :+ " ++ show (
snd (head (gprime n))) ++ "i"
Prelude> gprime' 5
"1 :+ 2i"
```

これですっきりしたね。さて、show の説明をしておこう。show は引数で指定した値を文字列で表示するものだ。gprime が表示するのはリストであることに注意しよう。そのリストには、目的の複素数が実部と虚部のタプル形式で入っている。われわれが欲しいのは実部 a と虚部 b が $a+bi$ の形で示されたものである。そこでリストからタプル形式だけ取り出すのが head の役目だ。

タプルのひとつ目が実部の値、2 つ目が虚部の値であるから、fst と snd で取り出せる。よって、あとはそれを上手に加工すればよい。show は文字列を返すので、それら文字列には自由に他の文

字列を付け足すことができる。ここでは虚部に `i` を付けて、実部と`:+`でつなげば完成だ。`+`でなく`:+`にしたのは **Haskell** 流を取り入れたかっただけである。

しかしながら、このスクリプトは完璧にほど遠い。実は、共役複素数の積に分解できない整数はエラーになってしまうので、それは回避したい。それに、単発で複素数の分解を表示しても大して面白くない。循環小数の循環でしたような、ガウス平面においても素数であり続ける**ガウス素数**のリストを見たいものだ。