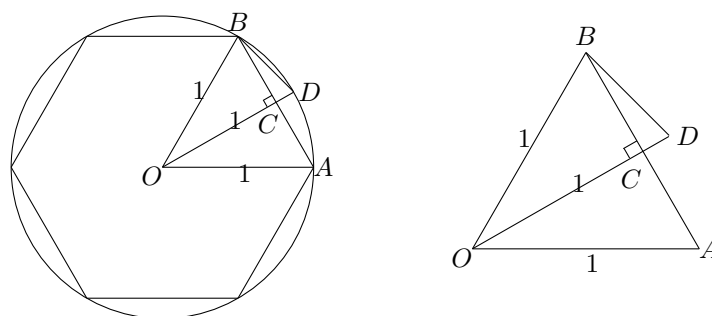


3 分割を細かく...

半径 1 の円に内接する正 6 角形と外接する正 6 角形を用いた円周率の計算では、たいした成果を得られなかった。最大の原因は、内側と外側の多角形が粗いためと思われる。もっと細かい分割がしたい。それには、辺の数を倍々にしていけばよいだろう。ただし、それには少し戦略を立てなくてはならない。



図は、内接する多角形の辺の数を倍にする際の様子を一部拡大したものだ。すべての線を描いてないが、辺 AB を分割して AD と DB に分けたところである。図には直角三角形が散見されるので、三平方の定理を当てはめながら考えてみよう。まず、 $\triangle OCB$ に対して $OC^2 + CB^2 = OB^2$ であるから

$$OC = \sqrt{OB^2 - CB^2}$$

が分かる。 $\triangle BCD$ に対しても同様に

$$CD = \sqrt{BD^2 - CB^2}$$

が分かる。そして、 OD は円の半径で 1 であるから、 $OC + CD = 1$ だ。よって、

$$\sqrt{OB^2 - CB^2} + \sqrt{BD^2 - CB^2} = 1$$

が導かれた。

さて、 OB は円の半径だから 1 である。 CB は AB の半分であるが、 AB は数を倍にする前の辺であったことに注意してほしい。そこで正 n 角形の一辺の長さという意味で $AB = l_n$ と書くことにすれば、 $CB = \frac{l_n}{2}$ である。次に、 BD は辺の数が倍の多角形の一辺だから、正 $2n$ 角形の一辺だ。そこで正 $2n$ 角形の一辺の長さという意味で $BD = l_{2n}$ と書くことにする。これより、先ほどの関係式は

$$\sqrt{1 - \left(\frac{l_n}{2}\right)^2} + \sqrt{l_{2n}^2 - \left(\frac{l_n}{2}\right)^2} = 1$$

と書き直すことができる。

なにやら複雑な式になっているが、 $\sqrt{1 - \left(\frac{l_n}{2}\right)^2}$ を移項して両辺を 2 乗したのち、再び $\sqrt{\quad}$ を消すために移項と 2 乗を施すと、意外にすっきりした式

$$l_{2n}^4 - 4l_{2n}^2 = -l_n^2$$

が現れる。

この式は l_{2n}^2 に関する 2 次方程式と見なせば、 l_{2n} について解くことが可能だ。こうすればよい。

$$\begin{aligned} l_{2n}^4 - 4l_{2n}^2 + 4 &= 4 - l_n^2 && \text{(両辺に 4 を足した)} \\ (l_{2n}^2 - 2)^2 &= 4 - l_n^2 && \text{(展開の公式より)} \\ l_{2n}^2 - 2 &= -\sqrt{4 - l_n^2} && \text{(平方根の定義より)} \\ l_{2n}^2 &= 2 - \sqrt{4 - l_n^2} && \text{(2 を移項)} \\ l_{2n} &= \sqrt{2 - \sqrt{4 - l_n^2}} && \text{(平方根の定義より)} \end{aligned}$$

3 行めで平方根を求める際、“+” でなく “-” にしたのを不思議に思うかもしれない。しかし、 $l_{2n} \leq 1$ であることに注意してもらいたい。 $l_{2n}^2 - 2$ は負の値なのだ。

一体、何が分かったんだろう。それは、 l_n の値が分かれば l_{2n} の値が計算できるということだ。はじめに私たちが考えた多角形は正六角形だから、 $l_6 = 1$ から始まっている。辺の数を 2 倍にすると正 12 角形ができる。その一辺の長さ l_{12} は、 $\sqrt{2 - \sqrt{4 - l_6^2}}$ を計算して $l_{12} = \sqrt{2 - \sqrt{4 - 1^2}} \approx 0.517638$ だ。正 12 角形の一辺を 6 倍したものが円周率となるので、3.105828 が分かる。電卓による計算とはいえ、なかなかいい感じだ。

こういった計算こそ PowerShell の出番である。

[ps script]

```
PS C:\Users\Yours > $l = 1; $n = 3; 1..15 | foreach {
>> $l = [math]::sqrt(2 - [math]::sqrt(4 - $l * $l)); $l * ($n *= 2)
>> }
>>
```

PowerShell について詳しい説明はあまりできないが、まったく言わないのでは旅の興をそぎかねない。ちょっとだけ解説しておこう。

まず、変数は “\$何とか” で表す。変数は宣言抜きで使えるため、最初の一辺の長さ \$l と多角形半周分の辺の数 \$n にいきなり初期値を与えた。; は文の区切りである。foreach{} 文は繰り返し計算に有効な文で、この旅ではよく使う。繰り返し範囲をパイプラインによって、先に 1..15 で与えている。入力する文が少々長いので、ここで一旦、キャリッジリターンキーを押しておく。するとプロンプトが “>> ” に変化するので、続きをここに入力する。2 行めの前の文が $l_{2n} = \sqrt{2 - \sqrt{4 - l_n^2}}$ であることは想像できるだろう。後ろの文は、\$n を 2 倍した上で \$l と掛けている。C や Java などではおなじみの表記だ。} で計算式の入力は終了だ。プロンプトが “>> ” となったところでキャリッジリターンキーをもう一度押すと、即座に計算結果が表示される。

[ps script]

3.10582854123025
3.13262861328124
3.13935020304687
3.14103195089053
3.14145247228534
3.14155760791162
3.14158389214894
3.14159046323676
3.14159210604305
3.14159251658815
3.14159261864079
3.14159264532122
3.14159264532122
3.14159264532122
3.14159264532122

計算では辺の数を 98304 まで増やしたことになる。そのことはすぐ確認できる。 $\$n$ と入力してみよう。

[ps script]

```
PS C:\Users\Yours > $n  
98304
```

半周分の辺の数が9万を超えるところまで計算したにもかかわらず、途中から同じ値の繰り返しになってしまった。それもそのはずで、辺の数を増やして精度を上げようとする試みは結構なのだが、辺の数が増えれば当然一辺の長さはどんどん短くなってしまう。するとコンピュータの計算では桁落ちが生じ、結局精度が落ちてしまうのだ。正しい値は 3.1415926 までである。残念ながら、これでは円周率を何万桁も計算するわけにはいかない。