算額（その146）解析解 岐阜県大垣市西外側町 大垣八幡神社（大垣八幡宮） 天保年間 http://www.wasan.jp/gifu/ogakihatiman.html キーワード：等脚台形，累円 #Julia #SymPy #算額 #和算 #数学

第一問　等脚台形に甲乙の 2 円を入れる。その隙間に順次，丙円，丁円，戊円...を入れていく。甲円の径と最後に入れた円（黒円と呼ぼう）の径が分かっているときに，乙円から黒円まで何個の円が入ったか。

この問題は見た目は違うが，90度時計回りに回転すると本質的には算額（その36）と同じものである。

甲円，乙円，丙円，丁円，戊円，... 黒円の半径を r_1，r_2，r_3，r_4，r_5，…，r_n とおく。

算法助術の公式41により $\sqrt{r_1 r_2} = \sqrt{r_1 r_3} + \sqrt{r_3 r_2}$ である。 この式は，算法助術の公式40 **共通接線の長さ** $= 2\sqrt{r_1 r_2}$ を適用して導かれる。

両辺を $\sqrt{r_1 r_2 r_3}$ で割れば，$\dfrac{1}{\sqrt{r_3}} = \dfrac{1}{\sqrt{r_2}} + \dfrac{1}{\sqrt{r_1}}$ である。

黒円 r_n まで式を書き下す。

$$
\begin{aligned}
\frac{1}{\sqrt{r_3}} &= \frac{1}{\sqrt{r_2}} + \frac{1}{\sqrt{r_1}} \\
\frac{1}{\sqrt{r_4}} &= \frac{1}{\sqrt{r_3}} + \frac{1}{\sqrt{r_1}} \\
\frac{1}{\sqrt{r_5}} &= \frac{1}{\sqrt{r_4}} + \frac{1}{\sqrt{r_1}} \\
&\vdots \\
\frac{1}{\sqrt{r_n}} &= \frac{1}{\sqrt{r_{n-1}}} + \frac{1}{\sqrt{r_1}}
\end{aligned}
$$

ここで，具体例も併せて考えるために，等脚台形内に甲円，乙円，丙円，丁円，戊円の 5 個の円が入っている場合を考える。

```
include("julia-source.txt")
using SymPy
@syms n, r1, r2, r3, r4, r5
eq3 = 1/√r3 == 1/√r2 + 1/√r1
eq4 = 1/√r4 == 1/√r3 + 1/√r1
eq5 = 1/√r5 == 1/√r4 + 1/√r1;
```

$r1$, r_5 が既知なので，eq3, eq4, eq5 の連立方程式を一度に解いて r_2, r_3, r_4 を求めることもできる。

```
res = solve((eq3, eq4, eq5), (r2, r3, r4))[1]
```

```
(r1^3*r5/((sqrt(r1) - sqrt(r1*r5/(-2*sqrt(r1)*sqrt(r5) + r1 + r5)))^2*(sqr
```

r_2 は r_1, r_5 のみで決まる。

```
# r2
res[1]
```

$$\frac{r_1^3 r_5}{\left(\sqrt{r_1}-\sqrt{\frac{r_1 r_5}{-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5}}\right)^2\left(\sqrt{r_1}-\sqrt{\frac{r_1^2 r_5}{\left(\sqrt{r_1}-\sqrt{\frac{r_1 r_5}{-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5}}\right)^2\left(-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5\right)}}\right)^2\left(-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5\right)}$$

```
# r3
res[2]
```

$$\frac{r_1^2 r_5}{\left(\sqrt{r_1}-\sqrt{\frac{r_1 r_5}{-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5}}\right)^2\left(-2\sqrt{r_1}\sqrt{r_5}+r_1+r_5\right)}$$

```
# r4
res[3]
```

$$\frac{r_1 r_5}{\left(\sqrt{r_1}-\sqrt{r_5}\right)^2}$$

それぞれの最後から 2 番目の円以外の式が複雑になるので，逆順に（eq5, eq4, eq2 の順に）解いて r_4, r_3, r_2 の順に求めてもよい。

```
ans_r4 = solve(eq5, r4)[1]
@show(ans_r4)
```

```
ans_r4 = r1*r5/(sqrt(r1) - sqrt(r5))^2
```

$$\frac{r_1 r_5}{\left(\sqrt{r_1}-\sqrt{r_5}\right)^2}$$

```
ans_r3 = solve(eq4, r3)[1]
@show(ans_r3)
```

```
ans_r3 = r1*r4/(sqrt(r1) - sqrt(r4))^2
```

$$\frac{r_1 r_4}{\left(\sqrt{r_1}-\sqrt{r_4}\right)^2}$$

```
ans_r2 = solve(eq3, r2)[1]
@show(ans_r2)
```

```
ans_r2 = r1*r3/(sqrt(r1) - sqrt(r3))^2
```

$$\frac{r_1 r_3}{\left(\sqrt{r_1}-\sqrt{r_3}\right)^2}$$

$r_1 = 225$, $r_5 = 10$ の場合を考える。

```
ans_r4(r1 => 225, r5 => 10).evalf()
```

16.0563275729879

```
ans_r3(r4 => ans_r4)(r1 => 225, r5 => 10).evalf()
```

29.8950324068

```
ans_r2(r3 => ans_r3)(r4 => ans_r4)(r1 => 225, r5 => 10).evalf()
```

74.0253073352042

これで言えることは，「乙円 r_2 の大きさは，r_1，r_5 つまり甲円と黒円の大きさにより，自動的に決定される」ということである。r_2 がこの値と異なれば，隣同士の円が外接しない（隙間ができたり重なったりする）。

術で「甲円の径と最後に入れた円の径が分かっているとき」という条件しかない（乙円についての記述がない）のは，「必要がない」のではなく，「（誤差範囲内で）自動的に決まってしまう」からである。

さて，本筋に戻ろう。$n-2$ 本の関係式において，辺々加えると対応する項が打ち消し合うものがあり，以下のように簡単な式になる。

$$\frac{1}{\sqrt{r_n}} = \frac{1}{\sqrt{r_2}} + \frac{n-2}{\sqrt{r_1}}$$

```
@syms n, r1, r2, rn
eq = 1/sqrt(rn) == 1/sqrt(r2) + (n - 2)/sqrt(r1)
```

$$\frac{1}{\sqrt{rn}} = \frac{1}{\sqrt{r_2}} + \frac{n-2}{\sqrt{r_1}}$$

r_1, r_n が既知なので，この関係式を解いて n を求める。

```
ans_n = solve(eq, n)[1]
@show(ans_n)
```

```
ans_n = sqrt(r1)/sqrt(rn) - sqrt(r1)/sqrt(r2) + 2
```

$$\frac{\sqrt{r_1}}{\sqrt{rn}} - \frac{\sqrt{r_1}}{\sqrt{r_2}} + 2$$

つまり，$n = \sqrt{\frac{r_1}{r_n}} - \sqrt{\frac{r_1}{r_2}} + 2$ である。繰り返すが，r_2 は何でもよいのではなく，r_1, r_n により，自動的に決まるのである。

上の数値例でいえば，「n は正確に 5」である。

```
ans_n(r1 => 225, rn => 10, r2 => 74.0253073352042).evalf()
```

5.0

時岡はここで，

$\sqrt{\frac{r_1}{r_n}} = \sqrt{\frac{r_1}{r_2}} + n - 2$ にガウス記号をとると，$\left\lfloor\sqrt{\frac{r_1}{r_n}}\right\rfloor = n + \left\lfloor\sqrt{\frac{r_1}{r_2}} - 2\right\rfloor$

と言い始め，

$\left\lfloor\sqrt{\frac{r_1}{r_n}}\right\rfloor = n$ となるのは，$\left\lfloor\sqrt{\frac{r_1}{r_2}} - 2\right\rfloor = 0$ i.e. $2 \leqq \sqrt{\frac{r_1}{r_2}} < 3$ のとき

としている。

一見正しそうであるが，$\sqrt{r_1/r_2}$ が 2 以上，3 未満というのは少し考えてもおかしい。制限が強すぎるし，何度も書いているが「r_2 は何でもよいのではなく，r_1, r_n により，自動的に決まるのである」。「r_2 が取りうる範囲」などない。

上の数値例でも，$\sqrt{r_1/r_2} = 1.743416490252569$ である。

```
# sqrt(r1/r2)
sqrt(225/74.0253073352042)
```

```
1.743416490252569
```

そもそも，円の個数 n は計算式で計算した数値の床(floor)を取るのであって，途中の項の床を取って和（差）を計算するものではない。

結論として，時岡は，「甲径/乙径の正の平方根が 2 以上 3 未満のときは，円の個数は $\lfloor\sqrt{甲径/黒径}\rfloor$」としている。

これによれば，上の数値例では「甲径/乙径の正の平方根が 2 以上 3 未満のとき」に該当しないので答えがないことになる。

補足として 1. 「甲径/乙径の正の平方根が 2 未満のときは，円の個数は $\lfloor\sqrt{甲径/黒径}\rfloor - 1$」 2. 「甲径/乙径の正の平方根が 3 以上 4 未満のときは，円の個数は $\lfloor\sqrt{甲径/黒径}\rfloor + 1$」 としている。

$r_1/r_2 = 1.743416490252569$ なので，前者が該当し，$n = 3$ となるが，不適切解である。

```
(floor(sqrt(r1/r5)) - 1)(r1 => 225, r5 => 10)
```

3

なお，術は無条件に「$\lfloor\sqrt{甲径/黒径}\rfloor$」なので 4 になるが，これも不適切解である。

```
sqrt(225/10), floor(sqrt(225/10))
```

```
(4.743416490252569, 4.0)
```

```
function func(r1, r5)
    r4 = r1*r5/(sqrt(r1) - sqrt(r5))^2
    r3 = r1*r4/(sqrt(r1) - sqrt(r4))^2
    r2 = r1*r3/(sqrt(r1) - sqrt(r3))^2
    n = sqrt(r1)/sqrt(r5) - sqrt(r1)/sqrt(r2) + 2
    @printf("r1 = %g, r2 = %g, r3 = %g, r4 = %g, r5 = %g\n", r1, r2, r3, r
    if 2 <= sqrt(r1/r2) < 3
        n2 = floor(sqrt(r1/r5))
    elseif sqrt(r1/r2) < 2
        n2 = floor(sqrt(r1/r5)) - 1
    elseif 3 <= sqrt(r1/r2) < 4
        n2 = floor(sqrt(r1/r5)) + 1
    else
        n2 = missing
    end
    @printf("n = %g,  sqrt(r1/r2) = %g, 時岡のn = %g\n", n, sqrt(r1/r2), n2
end;
func(225, 10)
```

```
r1 = 225, r2 = 74.0253, r3 = 29.895, r4 = 16.0563, r5 = 10
n = 5,  sqrt(r1/r2) = 1.74342, 時岡のn = 3
```

```
func(225, 15)
```

```
r1 = 225, r2 = 295.237, r3 = 64.1379, r4 = 27.2594, r5 = 15
n = 5,  sqrt(r1/r2) = 0.872983, 時岡のn = 2
```

$r_1 = 225$, $r_2 = 5$ のとき，時岡の式では $3 <= \sqrt{r_1/r_2} < 4$ に該当し，$n = 7$ になってしまう。これも不適切解である（正しくは $n = 5$）。

```
func(225, 5)
```

```
r1 = 225, r2 = 16.3627, r3 = 10.1501, r4 = 6.90532, r5 = 5
n = 5,  sqrt(r1/r2) = 3.7082, 時岡のn = 7
```

術では一律に，「円の個数は $\left\lfloor\sqrt{\frac{r_1}{r_n}}\right\rfloor$」なのでそれぞれ $n = 4$，$n = 6$ になってしまう。

```
sqrt(225/10), floor(sqrt(225/10))
```

```
(4.743416490252569, 4.0)
```

```
sqrt(225/5), floor(sqrt(225/5))
```

```
(6.708203932499369, 6.0)
```

ちょっと考えてみよう。「甲円と黒円の大きさが与えられる」という状況で，乙円の大きさを適当に決めて丙円，丁円を順次描いていくとき，乙円の大きさがいい加減に決められたものであれば，最後の黒円は隙間にピッタリ入る保証はない。また，甲円と乙円を先に描き，まず黒円を先に描いてから逆順で円を描いていくと，最後に丙円が隙間にぴったり入る保証もない。

つまり，先に求めた $n = \sqrt{\frac{r_1}{r_n}} - \sqrt{\frac{r_1}{r_2}} + 2$ の式の中に乙円の大きさ r_2 が入っているのはそのためである。計算結果で得られた n を整数にするために floor をとるのは，便宜的なものである（誤差の影響を除くのならば，切り捨てではなく，四捨五入をすべきである）。すべての円が外接すべきものはすべて外接するためには，r_1，r_2，r_n がきっちり決まっており，計算すると整数の n が求まるのだ。r_1，r_2，r_n のどれか一つでも不正確，不適切な数であるときは，結果の n の floor を取って解を得たつもりになっても，その条件では所与の図は描けない。

問では，「隙間に順次，丙円，丁円，戊円...を入れていく」と描いてあるだけである。途中の円は隣同士互いに接し，また台形の斜辺にも接するように描かれるのであろうが，最後に描かれる円が一つ前の円とは接するのであろうが，台形の斜辺にも接するかどうかは記載がないというのが逃げ道であろう。

別法として，甲円と黒円の半径が与えられたとき，黒円をスタートとして，一つ前の円の半径を数値計算するプログラムを書くという方法がある。 甲円の半径と中心座標を r_1，$(0,\ r_1)$ 黒円の半径と中心座標を r_n，$(2\sqrt{r_1 r_n},\ r_n)$ とすれば， 黒円より一つ前の円の半径は，
$r_{n-1} = \frac{r_1 r_n}{r_1 - 2\sqrt{r_1 r_n} + r_n}$ 中心座標は $(2\sqrt{r_1 r_{n-1}},\ r_{n-1})$ である。

```
# 甲円の半径を r1 として，半径 rn の円の次に大きい円の半径を求める関数
nextcircle(rn, r1) = r1*rn/(r1 - 2sqrt(r1*rn) + rn)
```

```
nextcircle (generic function with 1 method)
```

最初にあげた数値例（甲円の半径 = 225, 黒円の半径 = 10）では以下のようになり，半径が 74.0253 の円は乙円にあたる。 半径が 407.1159688 の円は甲円より大きくなり，台形の上底と下底の大小関係が逆転する。 その次の 半径が 3417.6299 の円は更に大きいが，もはやこれらの円は台形の中に存在しない。 その後は この2つの円が交互に計算されるようになる。

```
r1 = BigFloat(225)
r = 10
for i = 1:10
    println(i, " ", r)
    r = nextcircle(r1, r)
end
```

```
1 10
2 16.056327572987898429636026268235987042208111885439743338299222878604653
3 29.895032406800035349009514346191877342648354822758342487239244853966406
4 74.025307335204214799987706049252428152439501548057964742861165031028827
5 407.115968815494063820047634792150060143024825471670661124621611894437 98
6 3417.6299364909248392794025139936680082085597752356170865030526205080009
7 407.11596881549406382004763479215006014302482547167066112462161189443798
8 3417.6299364909248392794025139936680082085597752356170865030526205080009
9 407.11596881549406382004763479215006014302482547167066112462161189443798
10 3417.6299364909248392794025139936680082085597752356170865030526205080000
```

甲円の半径 = 225, 黒円の半径 = 1 のとき，乙円までの半径は以下のようになる。乙円は黒円から数えて 15 番目で，その半径は甲円の半径と同じになる。

```
r1 = BigFloat(225)
r = 1
for i = 1:16
    println(i, " ", r)
    r = nextcircle(r1, r)
end
```

```
1 1
2 1.1479591836734693877551020408163265306122448979591836734693877551020408
3 1.3313609467455621301775147928994082840236686390532544378698224852071005
4 1.5625
5 1.8595041322314049586776859504132231404958677685950413223140495867768595
6 2.25
7 2.7777777777777777777777777777777777777777777777777777777777777777777777
8 3.515625
9 4.5918367346938775510204081632653061224489795918367346938775510204081632
10 6.25
11 9.0
12 14.0625
13 25.0
14 56.25
15 225.0
16 Inf
```

たとえば，$r_1 = 225$，$rn = 9$ の場合も同じである。

```
r1 = BigFloat(225)
r = 9
for i = 1:5
    println(i, " ", r)
    r = nextcircle(r1, r)
end
```

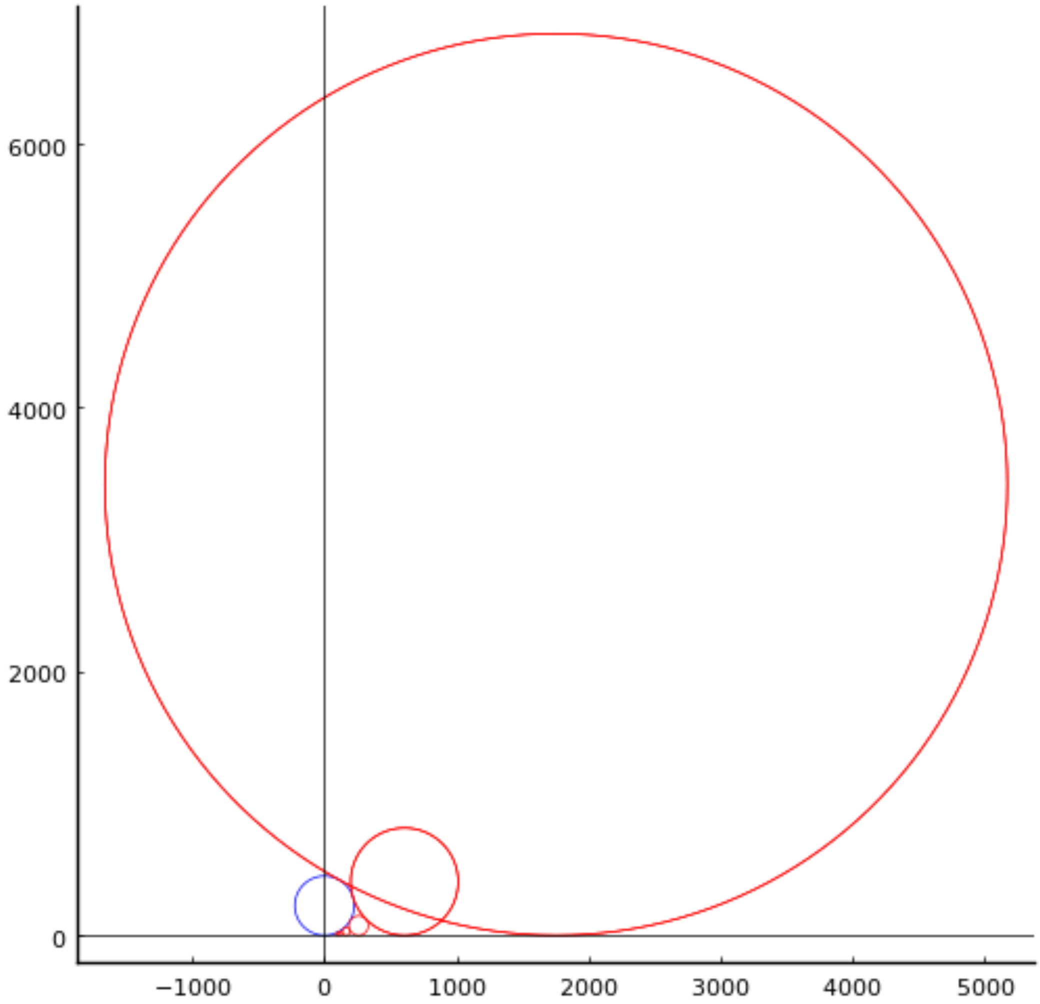
```
1 9
2 14.0625
3 25.0
4 56.25
5 225.0
```

以下のプログラムでは，乙円から黒円までの個数を数えているので，甲円も数えるなら，+1 する。

```
function draw(R, r, maxn, more)
    pyplot(size=(500, 500), grid=false, aspectratio=1, label="", fontfamil
    plot()
    println("個数 = ", floor(Int, sqrt(R/r)))
    circle(0, R, R, :blue)
    x = 2*sqrt(R)*sqrt(r) # x1
    for i = 1:maxn
        @printf("i = %d, x = %g, r = %g\n", i, x, r)
        circle(x, r, r, :red)
        r = nextcircle(r, R)
        x = 2sqrt(r1*r)
        #round(r, digits=10) > R && break
    end
    if more
        vline!([0], color=:black, lw=0.5)
    else
        plot!(showaxis=false)
    end
    hline!([0], color=:black, lw=0.5)
end;
```

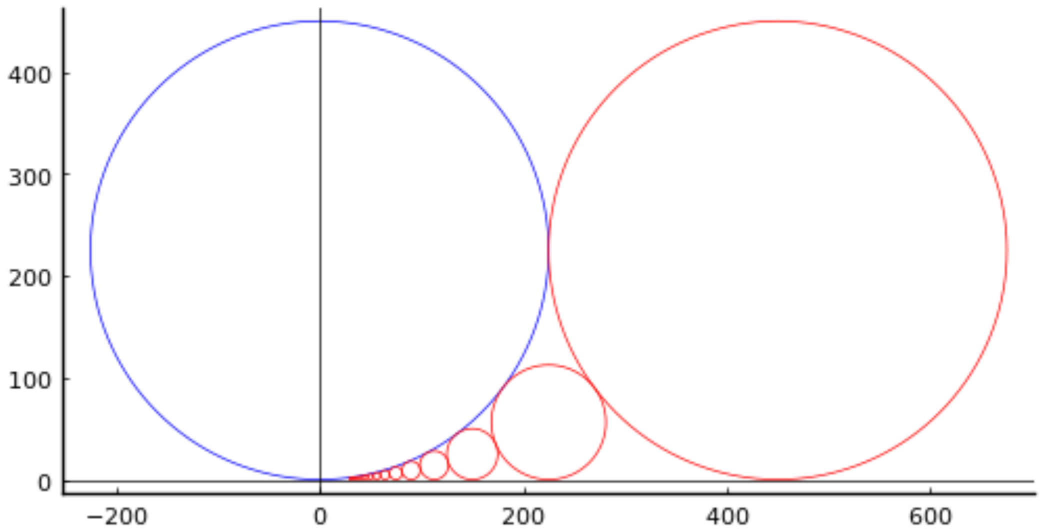
```
draw(225, 10, 8, true)
savefig("/Users/aoki/Downloads/fig1.png");
```

```
個数 = 4
i = 1, x = 94.8683, r = 10
i = 2, x = 120.211, r = 16.0563
i = 3, x = 164.029, r = 29.895
i = 4, x = 258.114, r = 74.0253
i = 5, x = 605.313, r = 407.116
i = 6, x = 1753.81, r = 3417.63
i = 7, x = 605.313, r = 407.116
i = 8, x = 1753.81, r = 3417.63
```




```
draw(225, 1, 15, true)
savefig("/Users/aoki/Downloads/fig1.png");
```

```
個数 = 15
i = 1, x = 30, r = 1
i = 2, x = 32.1429, r = 1.14796
i = 3, x = 34.6154, r = 1.33136
i = 4, x = 37.5, r = 1.5625
i = 5, x = 40.9091, r = 1.8595
i = 6, x = 45, r = 2.25
i = 7, x = 50, r = 2.77778
i = 8, x = 56.25, r = 3.51563
i = 9, x = 64.2857, r = 4.59184
i = 10, x = 75, r = 6.25
i = 11, x = 90, r = 9
i = 12, x = 112.5, r = 14.0625
i = 13, x = 150, r = 25
i = 14, x = 225, r = 56.25
i = 15, x = 450, r = 225
```




```
draw(225, 9, 5, true)
savefig("/Users/aoki/Downloads/fig3.png");
```

```
個数 = 5
i = 1, x = 90, r = 9
i = 2, x = 112.5, r = 14.0625
i = 3, x = 150, r = 25
i = 4, x = 225, r = 56.25
i = 5, x = 450, r = 225
```