

# Rexx for Fun

Walter Pachl 16 May 2023

- .Triangle Arithmetic
- .Turning the numbers into a picture (BMP)
- .Presentation with HTML
- .Using JDOR to create the picture
- .Puzzle Illustrations using JDOR
  - Area of a Pentagon
  - Size of Some Circles
- .Fraction Arithmetic

# The TRIANGLE and related Classes

Class	Attributes	Methods
triangle	E S M	init string (V E M in English)
point	x y	init string
line	A B	init string length k d kxd
circle	M r	init string

## Routines

ss(g)	Compute the perpendicular bisector of a line segment
cs(c,g)	Compute the crossing points of circle c and line g
sp(g,h)	Compute the crossing point of the lines g and h
normale(P,g)	Compute the normale from point P to line g
ws(A,B,C)	Compute the angular symmetric from point A
distpp(A,B)	Compute the distance between the points A and B
distpg(A,g)	Compute the distance from a point to a line
cc(c1,c2)	Compute the crossing points of circles c1 and c2
round(x)	Round a number to 3 decimal digits
qgl(a,b,c)	Solve a quadratic equation

# The CIRCLE Class

```
::CLASS circle public inherit stringlike
::ATTRIBUTE M
::ATTRIBUTE r
::METHOD init PUBLIC
  Expose M r
  Use strict Arg A,B,C=.nil
  Call tri_dbg 'circle~new'
  If datatype(B)='NUM' Then Do --assume M and r were supplied as arguments
    M=A -- center
    r=B -- radius
  End
Else Do
  g=.line~new(A,B) -- one side of the triangle
  h=.line~new(A,C) -- another side of the triangle
  M=sp(ss(g),ss(h)) -- cut their perpendicular bisectors
  If M=.nil Then
    Say a',' b',' C 'don't define a circle'
  Else
    r=distpp(A,M) -- radius
  End
End
```

# Computing the Inscribed Circle

Computing the Angle-bisectors

```
wsa=ws(T~E[1],T~E[3],T~E[2]); Say 'wsA' wsa~kxd
```

```
wsb=ws(T~E[2],T~E[1],T~E[3]); Say 'wsB' wsb~kxd
```

```
I=sp(wsa,wsb)
```

```
-- Area
```

```
a1=T~S[1]~length
```

```
b1=T~S[2]~length
```

```
c1=T~S[3]~length
```

```
s=(a1+b1+c1)/2
```

```
area=rxCalcsqrt(s*(s-a1)*(s-b1)*(s-c1))
```

```
rho=rxCalcsqrt((s-a1)*(s-b1)*(s-c1)/s)
```

```
Call o 'C b' I rho 'Inkreis'
```

# tric.rer produces this interface file

```
* sym 0 0 300 33 44 350
L s (300.000/33.000) (44.000/350.000) Seite a 407.462
L s (44.000/350.000) (0.000/0.000) Seite b 352.755
L s (0.000/0.000) (300.000/33.000) Seite c 301.810
L b (0.000/0.000) (181.962/179.164) Winkelsymmetrale a
L b (300.000/33.000) (18.723/148.932) Winkelsymmetrale b
...
S r 114.666667 127.666667 -70.666667 222.333333 300 33
C s (104.942/111.047) 104.734805 Feuerbachkreis
G m y=1.709*x-68.304 Eulersche Gerade
P p 114.666667 127.666667 Schwerpunkt
P g 75.7692084 61.1890138 Höhenschnittpunkt
P s (0.000/0.000) A
P s (300.000/33.000) B
P s (44.000/350.000) C
P b (112.150/110.425) Inkreismittelpunkt
```

# trid.rex creates a picture sym.bmp

*.Determine the width and height (m and n) of the picture*

*-Circumcircle*

*-Orthocenter*

*-Steiner Ellipsis*

# Initialize the picture data

```
hor=g.0m*8
ver=g.0n*8
g.0hor=hor
/* s is the header of the BMP file */
s='424d460000000000000000003600000002800000003800000002800000001001800000000000'x
s=s'1000000000000000000000000000000000000000000000'x
s=overlay(lend(hor),s,19,4)    /* insert horizontal size          */
s=overlay(lend(ver),s,23,4)    /* insert vertical size      */
If g.0zz Then Do
    zz.='**'copies(g.0white,hor)
    g.0zzlen=length(zz.)      /* initalize all lines in white */
End
Else
    z.=g.0white                /* initalize all pixels in white */

lend: Return reverse(d2c(arg(1),4))
--Convert an integer to the form needed in the BMP header
```

# Next fill the picture data with color. E.g. Circle

```
circle:
/*****
* Draw the circle and store the center
*****/
Parse Arg xm,ym,r,color
do w=0 To 360 By 0.5
    x=format(xm+r*rxcalccos(w,, 'D'),6,0)
    y=format(ym+r*rxcalcsin(w,, 'D'),6,0)
    Call ovl x,y,color
End
Call store format(xm,6,0) format(ym,6,0) color
Return

ovl: Procedure Expose z. zz. g. sigl 1
Parse Arg x,y,color
do xx=x-1 To x+1
    Do yy=y-1 To y+1
        call set yy,xx,color
    End
End
Return
```

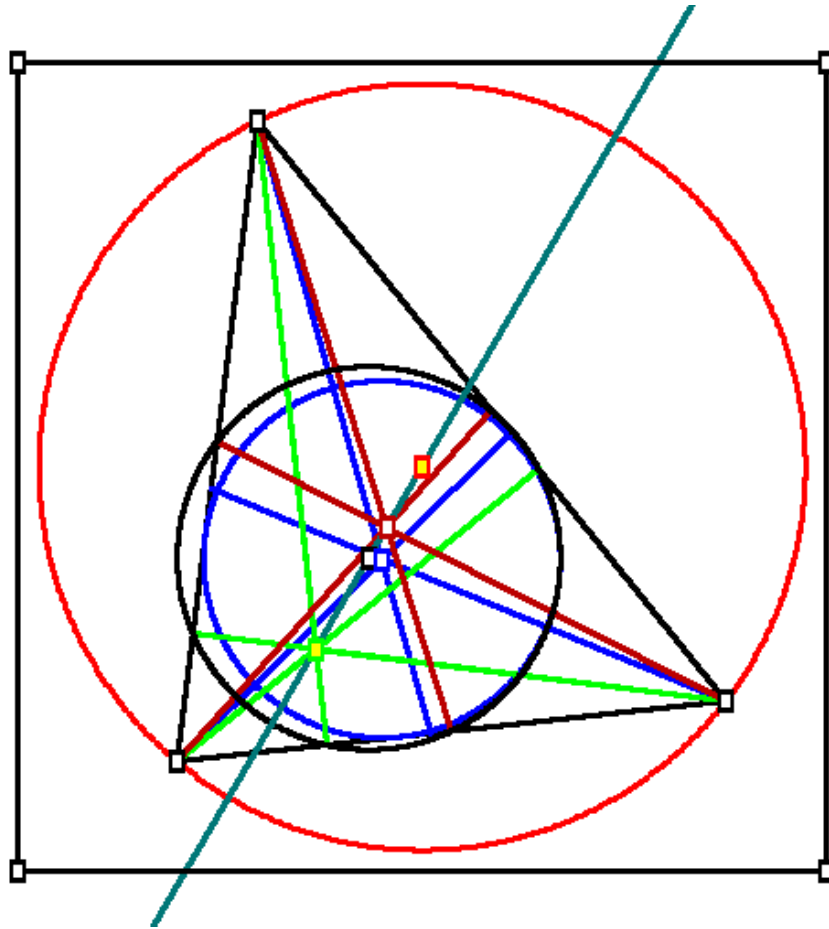


# Finally assemble the picture data and write the BMP file

```
Call charout pict,s

Do i=1 To z.0
  If g.0zz Then
    Call charout pict,substr(zz.i,3,g.0zzlen-2)
  Else Do
    Do j=1 To hor
      Call charout pict,z.i.j
    End
  End
End
Call lineout pict
```

And this is the created picture

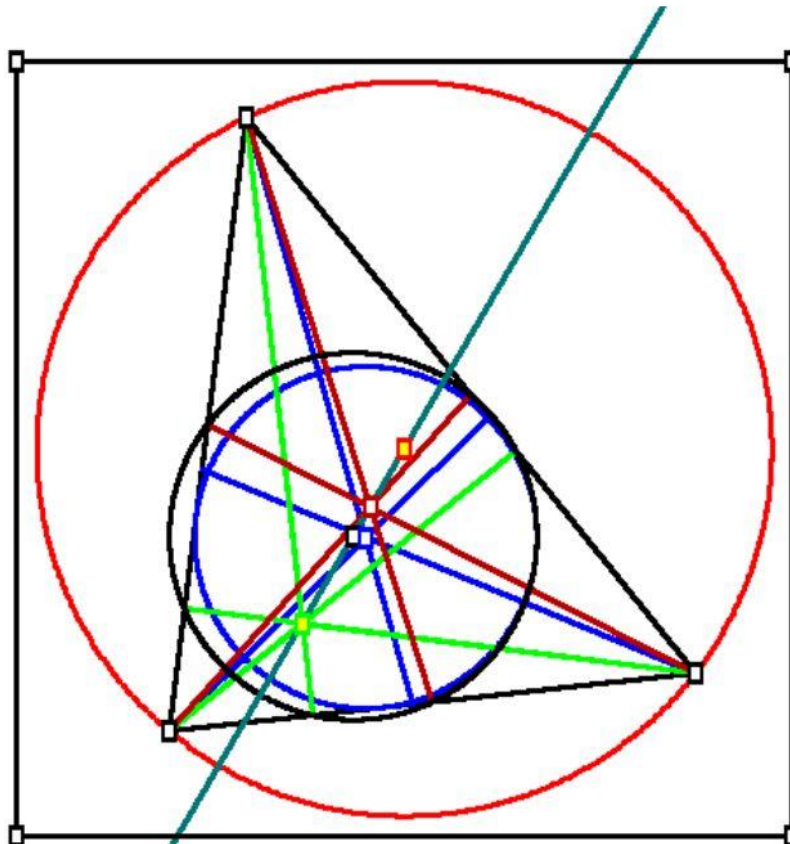


# The complete process

- .triset sets triresult as target
- .tric computes all numbers
- .trid creates the picture
- .trimerge creates html to show picture and numbers

# The final result

Compute and show data about Triangle sym: A (0/0) B (300/33) C (44/350)



Sides / Seiten:

BC  $a=407.462$  CA  $b=352.755$  AB  $c=301.810$

Perimeter / Umfang: 1062.030

Area / Fläche: 51774.000

Angles / Winkel:

$\alpha=76.557$   $\beta=57.354$   $\gamma=46.089$

Angle-bisectors / Winkelsymmetralen:

bsA:  $y=0.985 \cdot x$

bsB:  $y=-0.412 \cdot x+156.649$

bsC:  $y=-3.515 \cdot x+504.678$

Orthocenter / Höhenschnittpunkt: (75.769/61.189)

Circumcircle / Umkreis: (134.115/160.905)  $R=209.470$

Inscribed Circle / Inkreis: (112.150/110.425)  $r=97.500$

Steiner Ellipsis / Steiner Ellipse:  $a=238.422$ ,  $b=167.164$

Centroid / Schwerpunkt: (114.667/127.667)

Medians / Schwerlinien:

mA:  $y=1.113 \cdot x$

mB:  $y=-0.511 \cdot x+186.237$

mC:  $y=-3.146 \cdot x+488.434$

Feuerbach Circle / Feuerbachkreis: (104.942/111.047)  $r=104.735$

Euler's Line / Eulersche Gerade:  $y=1.709 \cdot x-68.304$

# Some Elements you may not know

## **.Euler's Line**

- Connects the centroid with the centers of**
- Circumcircle and Orthocenter**

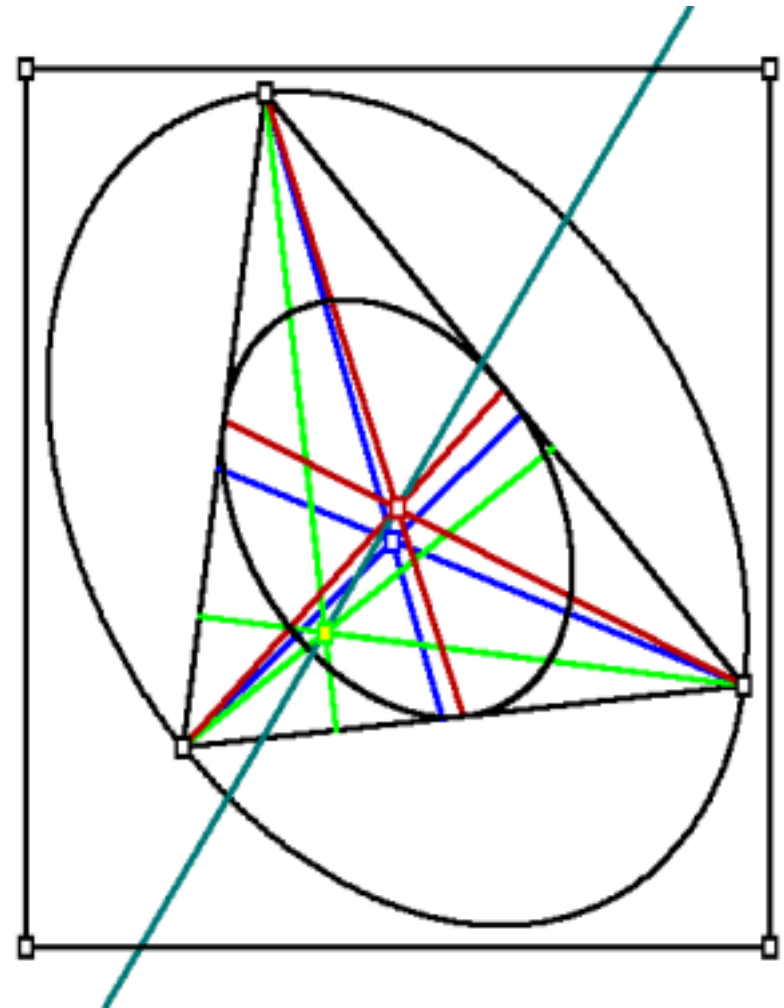
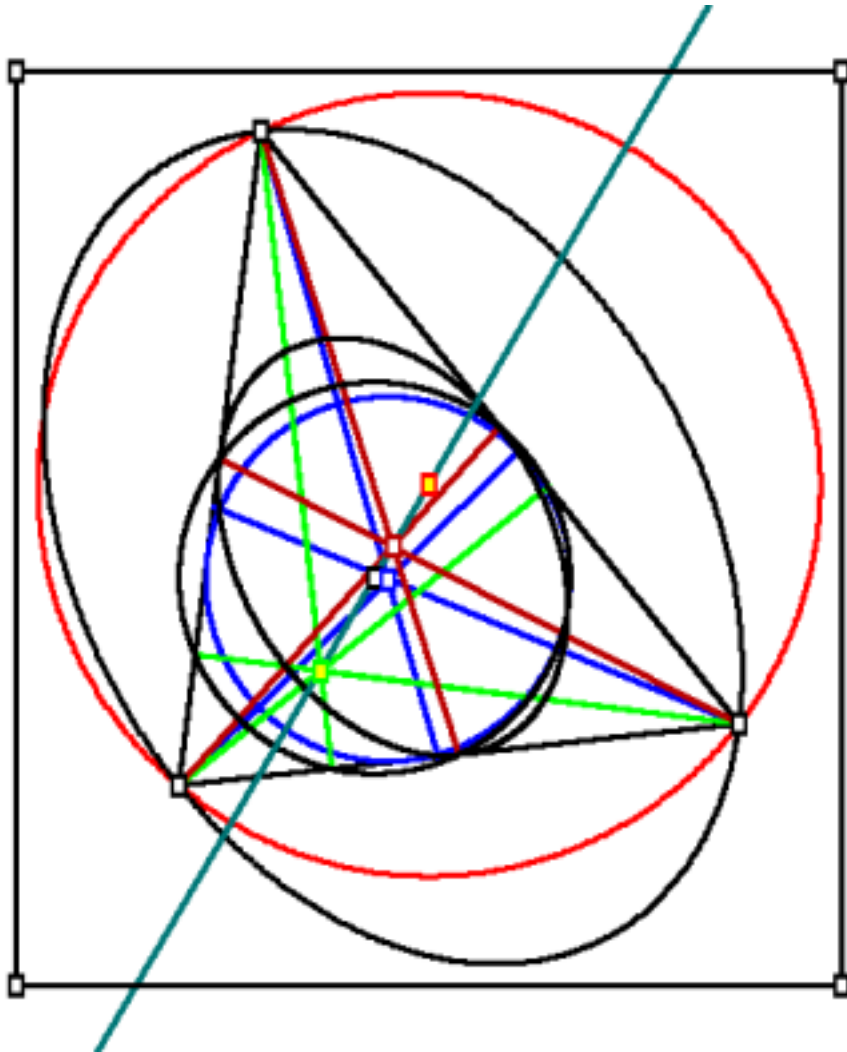
## **.Feuerbach Circle**

- Determined by the edges' midpoints**

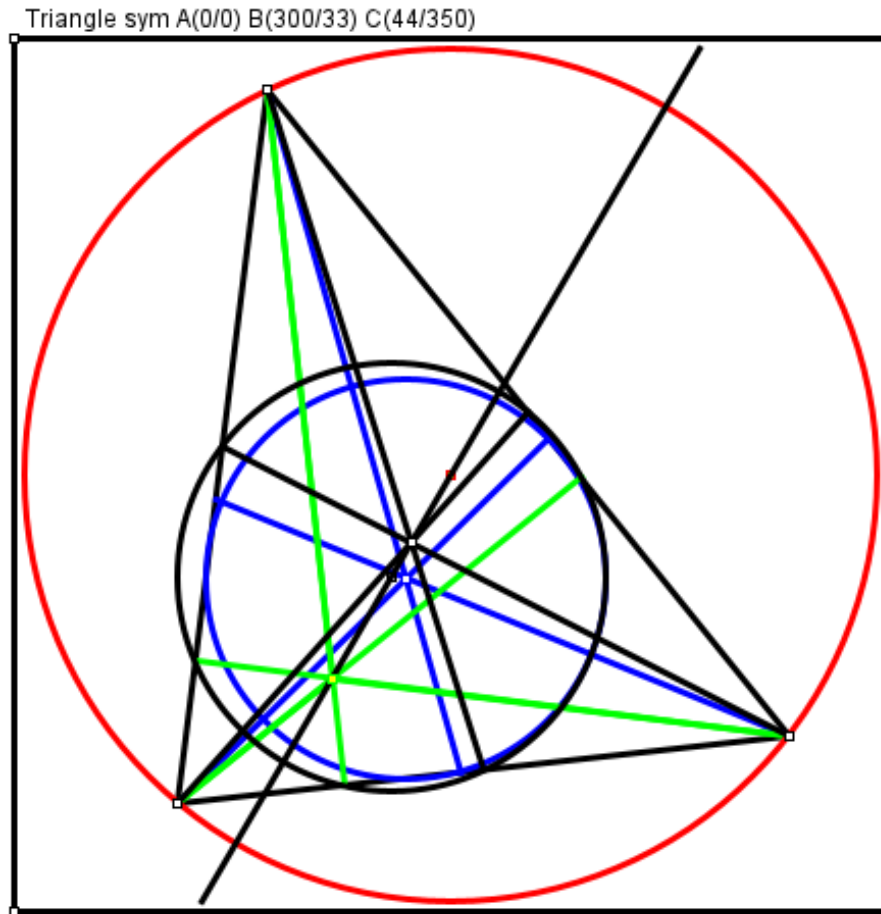
## **.Steiner Ellipsis**

- Determined by the triangle's vertices and having its center at the centroid**

# The Steiner Ellipsis

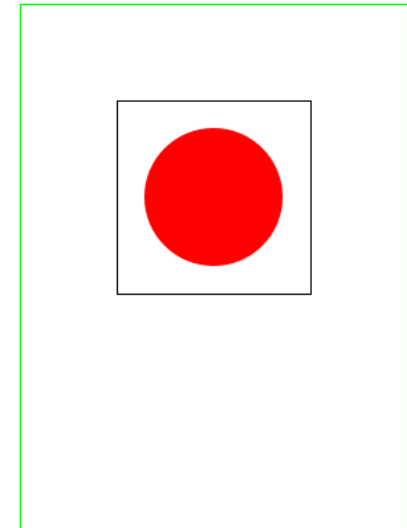


# The same Picture created using JDOR



# A Simple Program Using JDOR

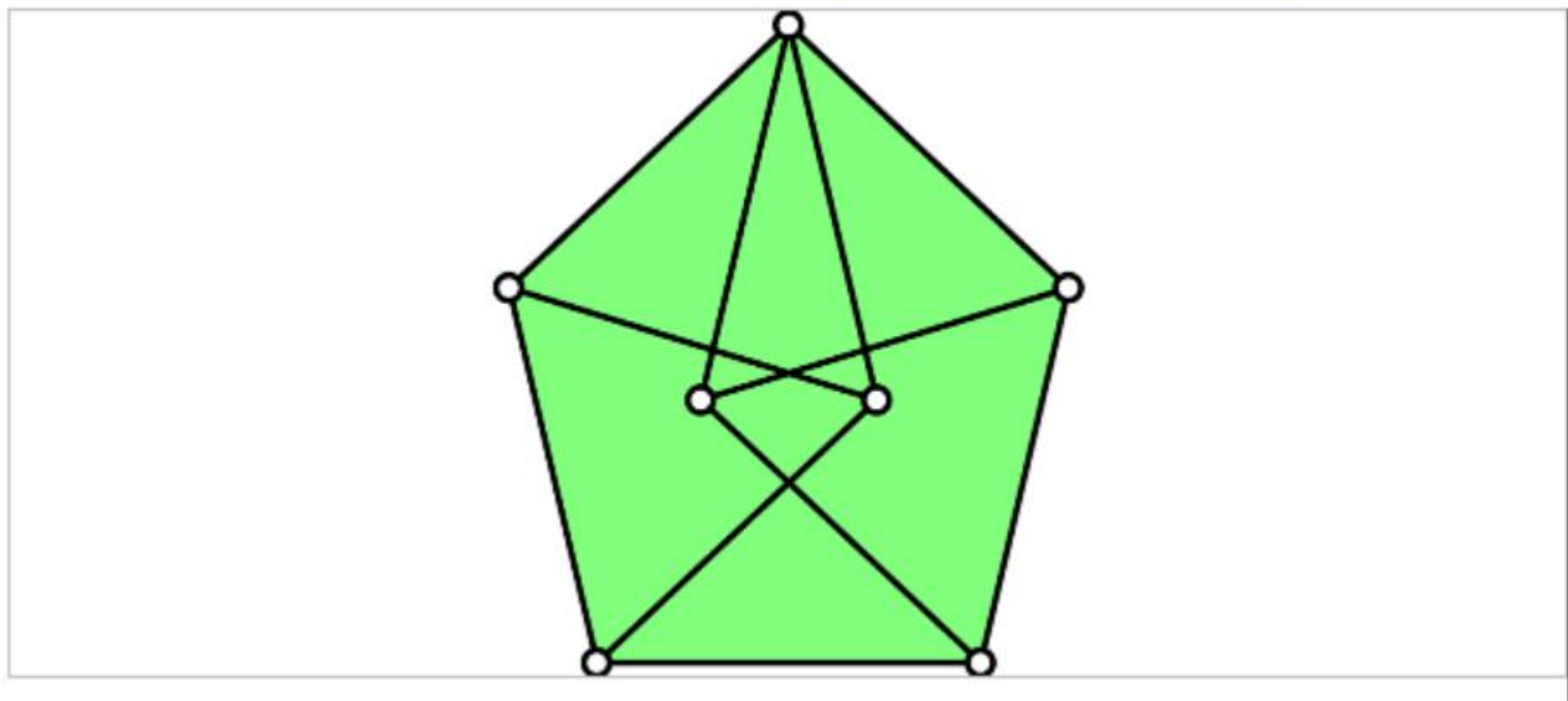
```
call BsfCommandHandler "add", "JDOR",,  
.bsf~new("org.oorexx.handlers.jdor.JavaDrawingHandler")  
address jdor          -- send commands to JDOR  
new 300 400           -- create new bitmap  
background white  
clearRect 300 400     -- paint it white  
goto 100 100; color red; fillOval 100 100; winShow; sleep 3  
goto 80 80; color black; drawrect 140 140;  
goto 10 10; color green; drawrect 280 380; winShow; sleep 3  
saveImage 'mini.png' -- save the current picture  
Exit  
::requires "BSF.CLS" -- get ooRexx-Java bridge
```





# Area of a Pentagon

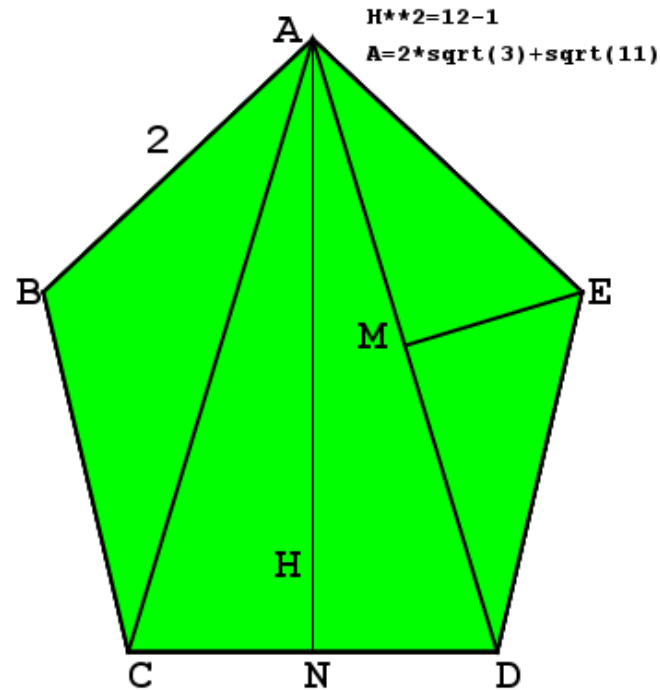
**Hemmes mathematische Rätsel: Wie lässt sich der Flächeninhalt bestimmen? von Heinrich Hemme**



© Heinrich Hemme (Ausschnitt)

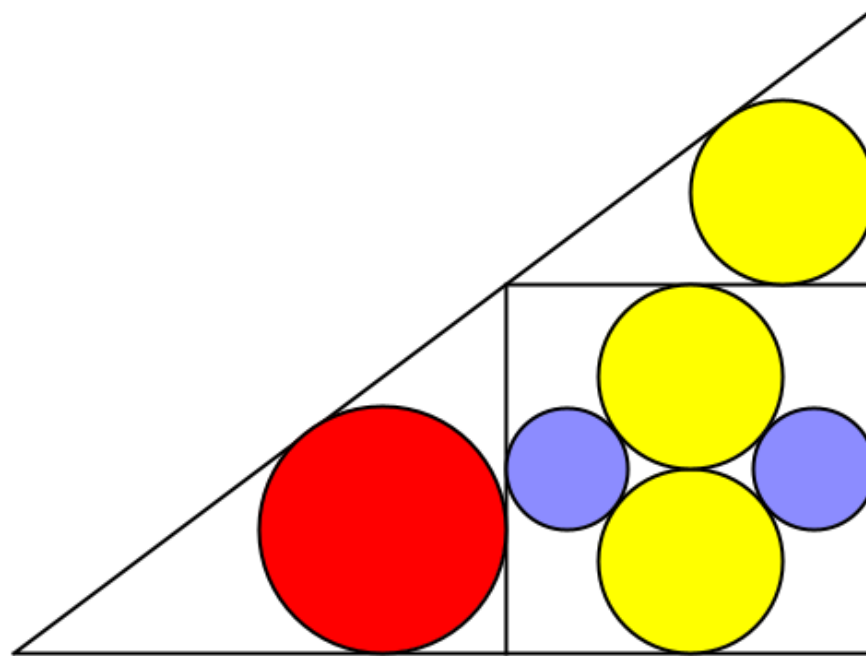
Alle elf Strecken in dieser Figur haben die Länge 2. Wie groß ist der Flächeninhalt des grünen Fünfecks?

# Area of the Pentagon: Final Picture



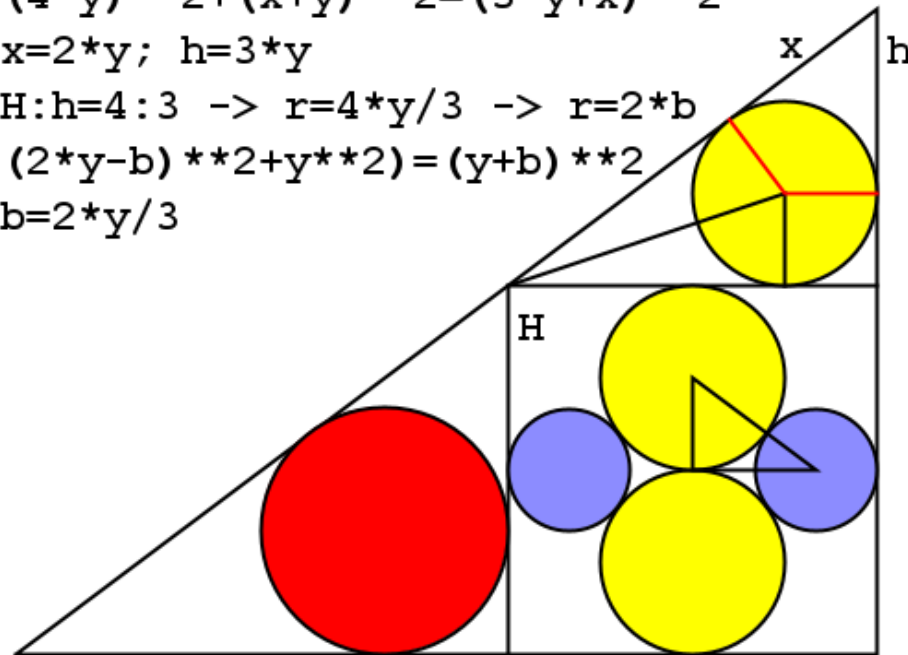
pentagon\_5.png

Red Radius is Blue Radius times 2



# Circles: The Final Picture

$$\begin{aligned}(4*y)**2+(x+y)**2 &= (3*y+x)**2 \\ x &= 2*y; \quad h = 3*y \\ H:h &= 4:3 \rightarrow r = 4*y/3 \rightarrow r = 2*b \\ (2*y-b)**2+y**2 &= (y+b)**2 \\ b &= 2*y/3\end{aligned}$$



3k\_9.png

# Fractions

**Welchen Wert hat dieser Bruch?**

$$\frac{1+1+1+1+1+1+1+1+1+1}{1+1+1+1+1+1+1+1+1+1}$$
$$\frac{1+1+1+1+1+1+1+1}{1+1+1+1+1+1+1+1}$$
$$\frac{1+1+1+1+1+1+1}{1+1+1+1+1+1+1}$$
$$\frac{1+1+1+1+1+1}{1+1+1+1+1+1}$$
$$\frac{1+1+1+1}{1+1+1+1}$$
$$\frac{1+1+1}{1+1+1}$$
$$\frac{1+1}{1+1}$$
$$\frac{1}{1}$$

© Heinrich Hemme (Ausschnitt)

# The Value of this Fraction

```
say 'x=nom/x'  
x=.fraction~new(2,1)  
Do n=3 To 10  
  nom=.fraction~new(n,1)  
  x=nom/x  
  Say n x  
End
```

```
say 'x=n/x'  
x=.fraction~new(2,1)  
Do n=3 To 10  
  x=n/x  
  Say n x  
End  
::requires fraction.cls
```

```
x=nom/x  
3 3/2  
4 8/3  
5 15/8  
6 16/5  
7 35/16  
8 128/35  
9 315/128  
10 256/63  
x=n/x  
3 1.5  
4 2.66666667  
5 1.875  
6 3.2  
7 2.1875  
8 3.65714286  
9 2.4609375  
10 4.06349206
```

# Cat and Mouse

In a circular pond with 2m diameter swims a mouse.

A cat watches it from the border.

The cat can't swim.

She runs four times as fast as the mouse can swim.

On solid ground the mouse runs incredibly fast.

How can the mouse escape?

This I am told by Horst Wegscheider:

Originally published in Martin Gardner's Column  
in Scientific American Nov./Dez. 1965