

# Visualisasi dan Perhitungan Geometri dengan EMT

---

Euler menyediakan beberapa fungsi untuk melakukan visualisasi dan perhitungan geometri, baik secara numerik maupun analitik (seperti biasanya tentunya, menggunakan Maxima). Fungsi-fungsi untuk visualisasi dan perhitungan geometri tersebut disimpan di dalam file program "geometry.e", sehingga file tersebut harus dipanggil sebelum menggunakan fungsi-fungsi atau perintah-perintah untuk geometri.

```
>load geometry
```

```
Numerical and symbolic geometry.
```

## Fungsi-fungsi Geometri

---

Fungsi-fungsi untuk Menggambar Objek Geometri:

```
defaultd:=textheight()*1.5: nilai asli untuk parameter d  
setPlotrangle(x1,x2,y1,y2): menentukan rentang x dan y pada bidang
```

koordinat

```
setPlotRange(r): pusat bidang koordinat (0,0) dan batas-batas
```

sumbu-x dan y adalah -r sd r

```
plotPoint (P, "P"): menggambar titik P dan diberi label "P"  
plotSegment (A,B, "AB", d): menggambar ruas garis AB, diberi label
```

"AB" sejauh d

```
plotLine (g, "g", d): menggambar garis g diberi label "g" sejauh d  
plotCircle (c,"c",v,d): Menggambar lingkaran c dan diberi label "c"  
plotLabel (label, P, V, d): menuliskan label pada posisi P
```

Fungsi-fungsi Geometri Analitik (numerik maupun simbolik):

```
turn(v, phi): memutar vektor v sejauh phi  
turnLeft(v): memutar vektor v ke kiri  
turnRight(v): memutar vektor v ke kanan  
normalize(v): normal vektor v  
crossProduct(v, w): hasil kali silang vektor v dan w.  
lineThrough(A, B): garis melalui A dan B, hasilnya [a,b,c] sdh.
```

$ax+by=c$ .

`lineWithDirection(A,v)`: garis melalui A searah vektor  $v$   
`getLineDirection(g)`: vektor arah (gradien) garis  $g$   
`getNormal(g)`: vektor normal (tegak lurus) garis  $g$   
`getPointOnLine(g)`: titik pada garis  $g$   
`perpendicular(A, g)`: garis melalui A tegak lurus garis  $g$   
`parallel (A, g)`: garis melalui A sejajar garis  $g$   
`lineIntersection(g, h)`: titik potong garis  $g$  dan  $h$   
`projectToLine(A, g)`: proyeksi titik A pada garis  $g$   
`distance(A, B)`: jarak titik A dan B  
`distanceSquared(A, B)`: kuadrat jarak A dan B  
`quadrance(A, B)`: kuadrat jarak A dan B  
`areaTriangle(A, B, C)`: luas segitiga ABC  
`computeAngle(A, B, C)`: besar sudut  $\angle ABC$   
`angleBisector(A, B, C)`: garis bagi sudut  $\angle ABC$   
`circleWithCenter (A, r)`: lingkaran dengan pusat A dan jari-jari  $r$   
`getCircleCenter(c)`: pusat lingkaran  $c$   
`getCircleRadius(c)`: jari-jari lingkaran  $c$   
`circleThrough(A,B,C)`: lingkaran melalui A, B, C  
`middlePerpendicular(A, B)`: titik tengah AB  
`lineCircleIntersections(g, c)`: titik potong garis  $g$  dan lingkaran  $c$   
`circleCircleIntersections (c1, c2)`: titik potong lingkaran  $c1$  dan

c2

`planeThrough(A, B, C)`: bidang melalui titik A, B, C

### Fungsi-fungsi Khusus Untuk Geometri Simbolik:

`getLineEquation (g,x,y)`: persamaan garis  $g$  dinyatakan dalam  $x$  dan  $y$   
`getHesseForm (g,x,y,A)`: bentuk Hesse garis  $g$  dinyatakan dalam  $x$  dan

$y$  dengan titik A pada

sisi positif (kanan/atas) garis  
`quad(A,B)`: kuadrat jarak AB  
`spread(a,b,c)`: Spread segitiga dengan panjang sisi-sisi  $a,b,c$ , yakni

$\sin(\alpha)^2$  dengan

$\alpha$  sudut yang menghadap sisi  $a$ .  
`crosslaw(a,b,c,sa)`: persamaan 3 quads dan 1 spread pada segitiga

dengan panjang sisi  $a, b, c$ .

`triplespread(sa,sb,sc)`: persamaan 3 spread  $sa,sb,sc$  yang membentuk

suatu segitiga

`doublespread(sa)`: Spread sudut rangkap Spread  $2\phi$ , dengan

$sa = \sin(\phi)^2 \text{ spread } a$ .

## Contoh 1: Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

```
>setPlotRange(-0.5,2.5,-0.5,2.5); // mendefinisikan bidang koordinat baru
```

Sekarang tentukan tiga titik dan gambarkan koordinat yang sudah tersedia.

```
>A=[1,0]; plotPoint(A,"A"); // definisi dan gambar tiga titik  
>B=[0,1]; plotPoint(B,"B");  
>C=[2,2]; plotPoint(C,"C");
```

Kemudian tiga segmen/ruas garis.

```
>plotSegment(A,B,"c"); // c=AB  
>plotSegment(B,C,"a"); // a=BC  
>plotSegment(A,C,"b"); // b=AC
```

Fungsi geometri mencakup fungsi untuk membuat garis dan lingkaran. Format untuk garis adalah [a, b, c], yang merepresentasikan garis dengan persamaan  $ax+by=c$ .

```
>lineThrough(B,C) // garis yang melalui B dan C
```

```
[-1, 2, 2]
```

Hitung garis tegak lurus melalui A pada BC.

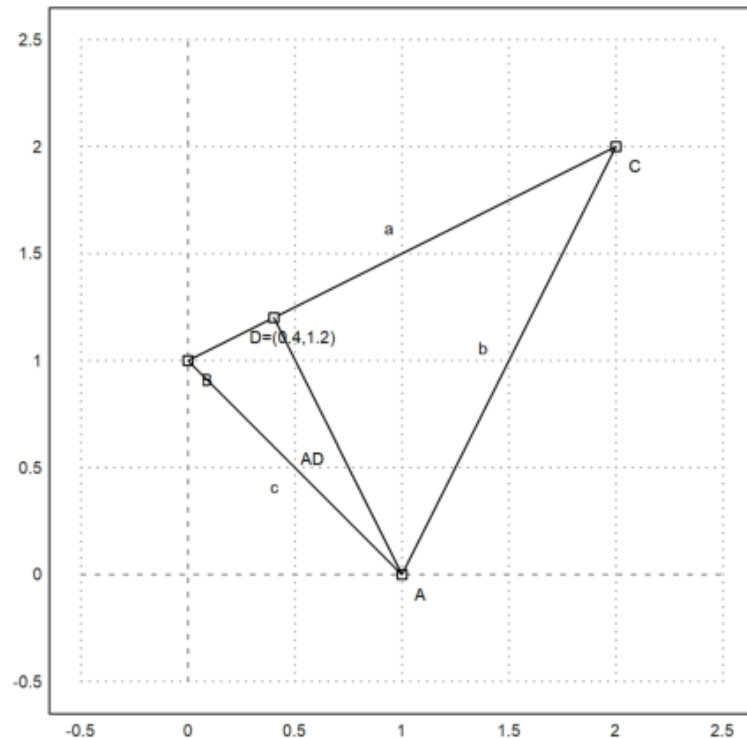
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan tentukan titik perpotongannya dengan BC.

```
>D=lineIntersection(h,lineThrough(B,C)); // D adalah titik potong h dan BC
```

Gambarkan plot.

```
>plotPoint(D,value=1); // koordinat D ditampilkan  
>aspect(1); plotSegment(A,D): // tampilkan semua gambar hasil plot...()
```



Hitung luas ABC:

$$L_{\triangle ABC} = \frac{1}{2}AD \cdot BC.$$

```
>norm(A-D)*norm(B-C)/2 // AD=norm(A-D), BC=norm(B-C)
```

1.5

Bandingkan dengan rumus determinan.

```
>areaTriangle(A,B,C) // hitung luas segitiga langsung dengan fungsi
```

1.5

Cara lain menghitung luas segitiga ABC:

```
>distance(A,D)*distance(B,C)/2
```

1.5

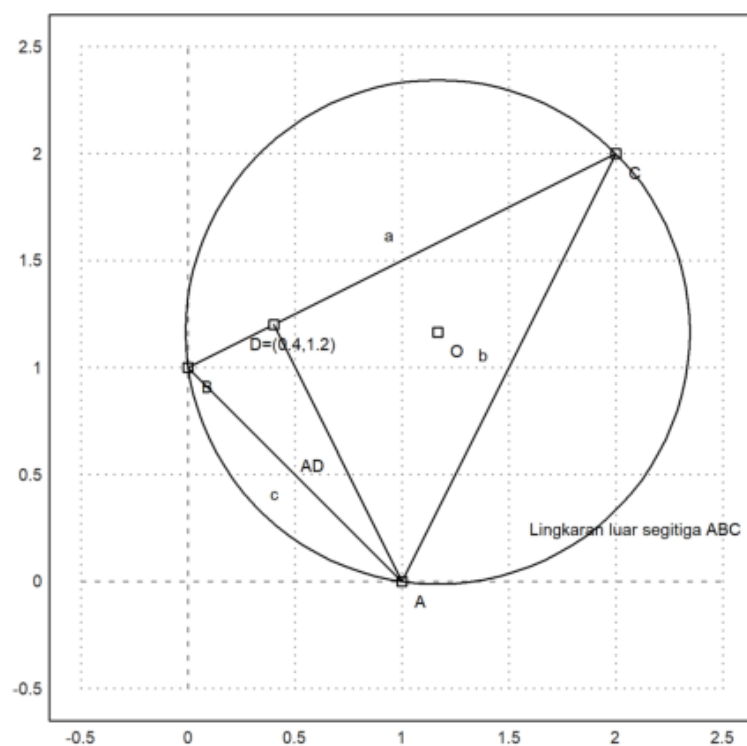
Sudut di titik C.

```
>degprint (computeAngle (B,C,A) )
```

36°52'11.63''

Sekarang lingkaran luar segitiga.

```
>c=circleThrough(A,B,C); // lingkaran luar segitiga ABC  
>R=getCircleRadius(c); // jari2 lingkaran luar  
>O=getCircleCenter(c); // titik pusat lingkaran c  
>plotPoint(O,"O"); // gambar titik "O"  
>plotCircle(c,"Lingkaran luar segitiga ABC"):
```



Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.

```
>O, R
```

[1.16667, 1.16667]  
1.17851130198

Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.

```
>l=angleBisector(A,C,B); // garis bagi <ACB  
>g=angleBisector(C,A,B); // garis bagi <CAB  
>P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
```

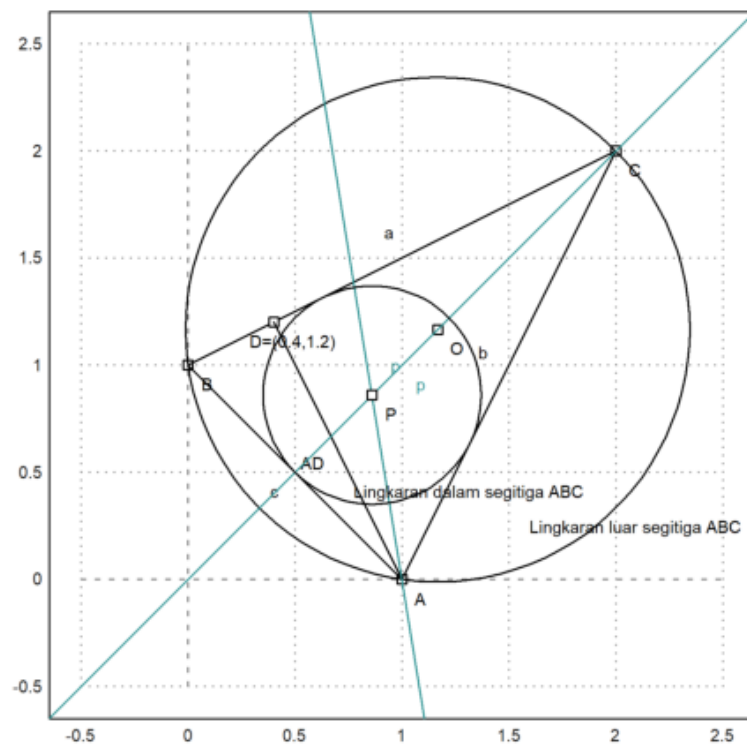
```
[0.86038, 0.86038]
```

Tambahkan semua titik dan garis ke dalam plot.

```
>color(5); plotLine(l); plotLine(g); color(1); // gambar kedua garis bagi sudut  
>plotPoint(P,"P"); // gambar titik potongnya  
>r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari lingkaran dalam
```

```
0.509653732104
```

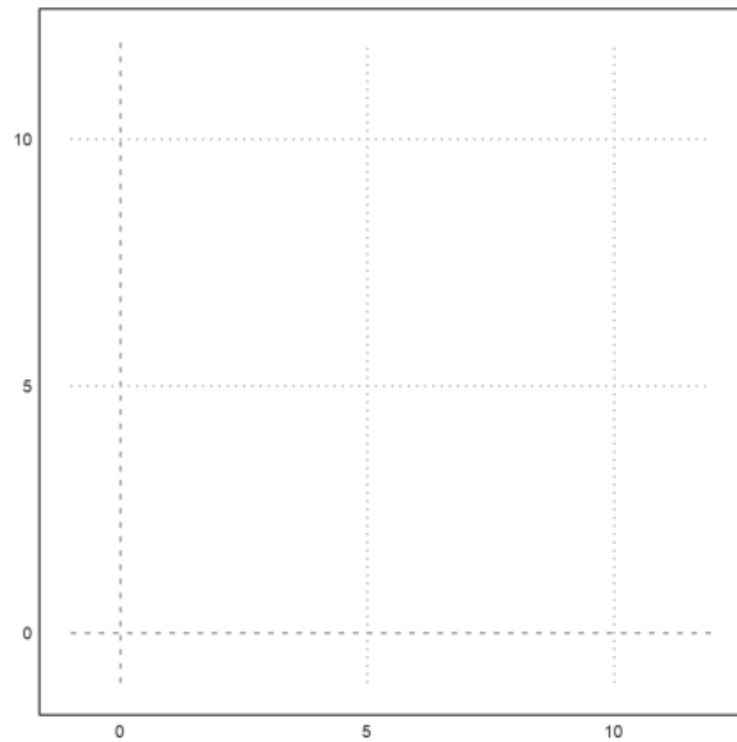
```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"): // gambar lingkaran dal
```



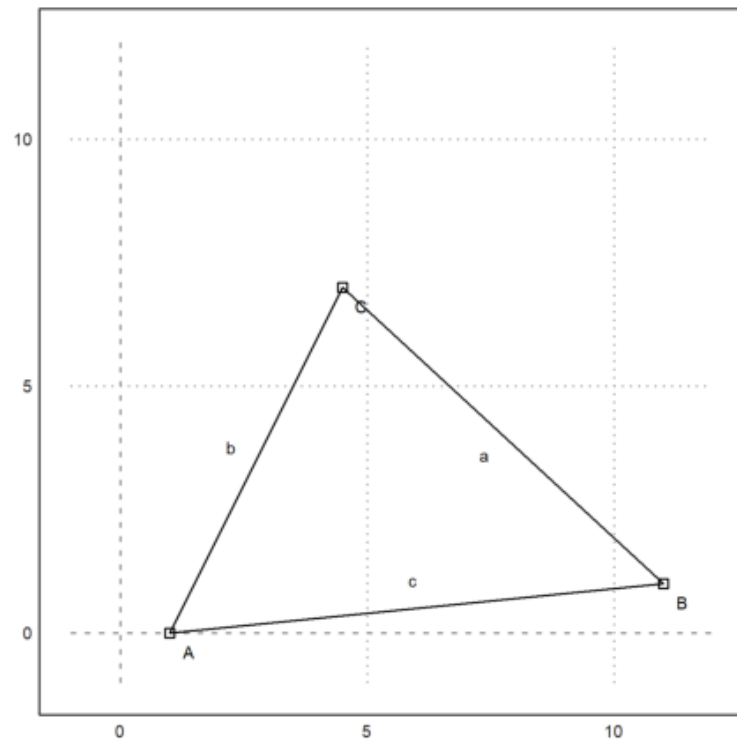
## Latihan

1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.

```
>setPlotRange(-1,12,-1,12):
```



```
>A=[1,0]; plotPoint(A,"A");  
>B=[11,1]; plotPoint(B,"B");  
>C=[4.5,7]; plotPoint(C,"C");  
>plotSegment(A,B,"c");  
>plotSegment(B,C,"a");  
>plotSegment(A,C,"b"):
```

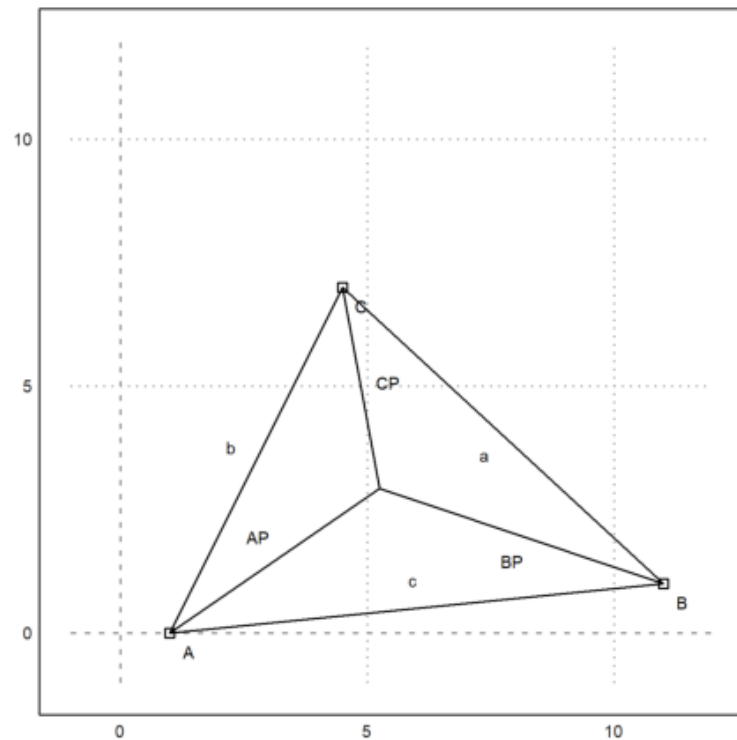


```
>l=angleBisector(A,C,B); // garis bagi <ACB
>g=angleBisector(C,A,B); // garis bagi <CAB
>i=angleBisector(A,B,C); // garis bagi <ABC
>P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
```

```
[5.24507, 2.9255]
```

```
>plotSegment(A,P); plotSegment(B,P); plotSegment(C,P):
```

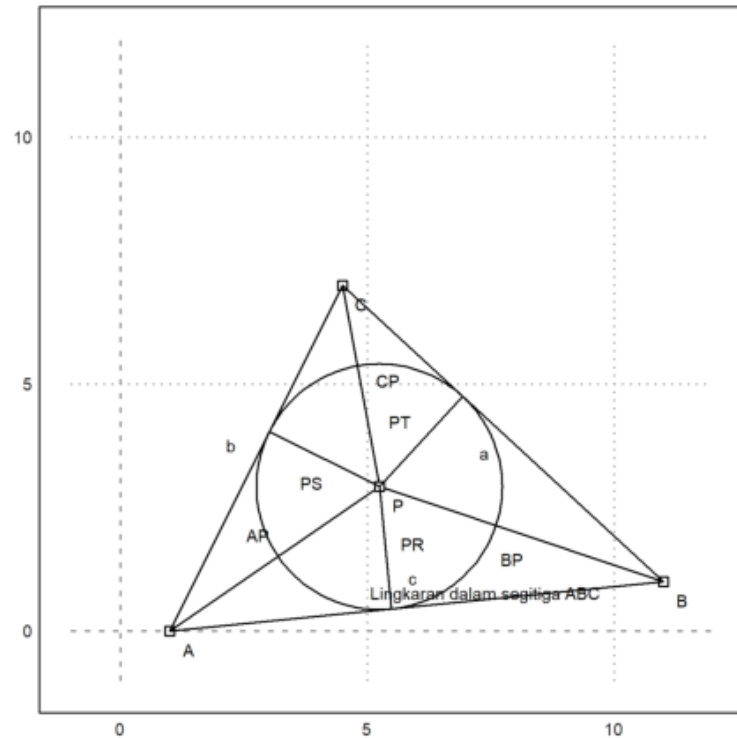




```
>plotPoint(P,"P"); // gambar titik potongnya
>r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari lingkaran dalam
```

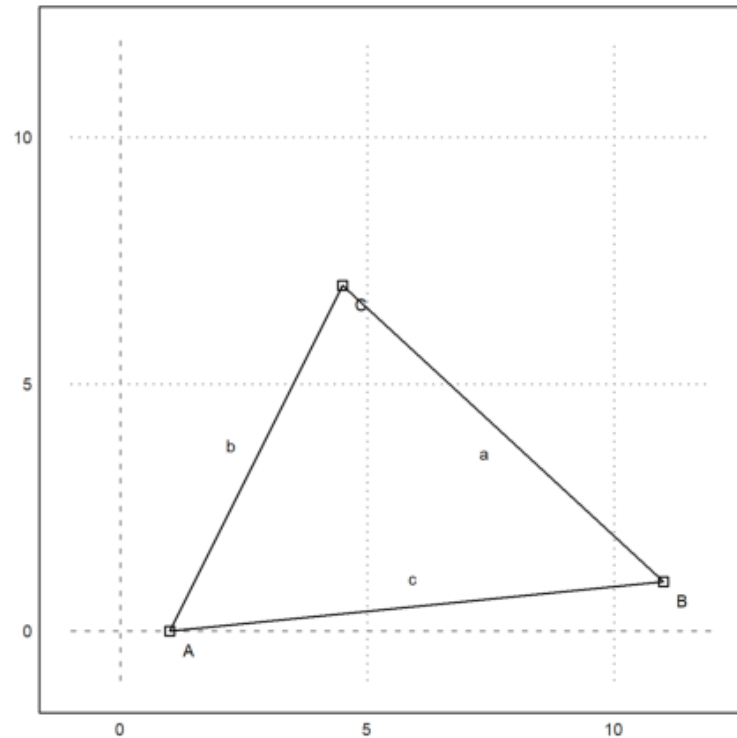
2.4885846425

```
>h=perpendicular(P,lineThrough(A,B));
>R=lineIntersection(h,lineThrough(A,B));
>plotSegment(P,R);
>j=perpendicular(P,lineThrough(A,C));
>S=lineIntersection(j,lineThrough(A,C));
>plotSegment(P,S);
>k=perpendicular(P,lineThrough(C,B));
>T=lineIntersection(k,lineThrough(C,B));
>plotSegment(P,T);
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"): // gambar lingkaran dal
```



2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?
3. Hitung luas segitiga tersebut.

```
>setPlotRange(-1,12,-1,12);
>A=[1,0]; plotPoint(A,"A");
>B=[11,1]; plotPoint(B,"B");
>C=[4.5,7]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b");
```



```
>areaTriangle(A,B,C) // Menghitung luas segitiga
```

33.25

4. Tunjukkan bahwa garis bagi sudut yang ke tiga juga melalui titik pusat lingkaran dalam.
5. Gambar jari-jari lingkaran dalam.

```
>setPlotRange(-1,12,-1,12);
>A=[1,0]; plotPoint(A,"A");
>B=[11,1]; plotPoint(B,"B");
>C=[4.5,7]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b");
>l=angleBisector(A,C,B); // garis bagi <ACB
>g=angleBisector(C,A,B); // garis bagi <CAB
>i=angleBisector(A,B,C); // garis bagi <CAB
>P=lineIntersection(l,g)
```

[5.24507, 2.9255]

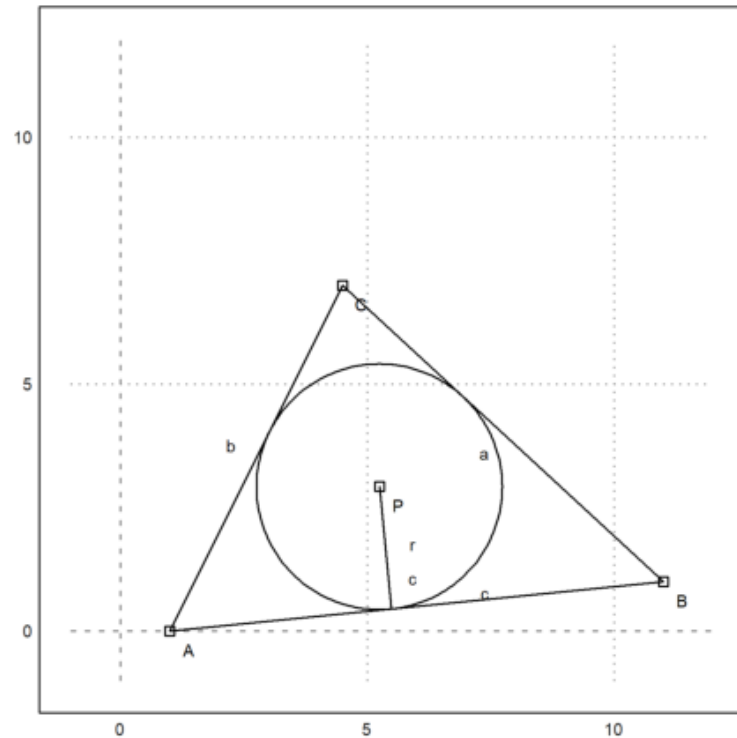
```
>plotPoint(P,"P");
>r=norm(P-projectToLine(P,lineThrough(A,B))) //jari-jari lingkaran dalam
```

2.4885846425

```

>h=perpendicular(P,lineThrough(A,B));
>D=lineIntersection(h,lineThrough(A,B));
>plotSegment(P,D,"r");
>plotCircle(circleWithCenter(P,r)):

```



6. Hitung luas lingkaran luar dan luas lingkaran dalam segitiga ABC. Adakah hubungan antara luas kedua lingkaran tersebut dengan luas segitiga ABC?

```

>setPlotRange(-5,12,-5,12);
>A=[1,0]; plotPoint(A,"A");
>B=[11,1]; plotPoint(B,"B");
>C=[4.5,7]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b");
>c=circleThrough(A,B,C); // lingkaran luar segitiga ABC
>R=getCircleRadius(c); // jari-jari lingkaran luar
>O=getCircleCenter(c); // titik pusat lingkaran c
>O, R // Titik pusat dan jari-jari lingkaran luar

```

```

[5.85526, 1.94737]
5.23123542767

```

```

>pi*r^2 // Luas lingkaran luar

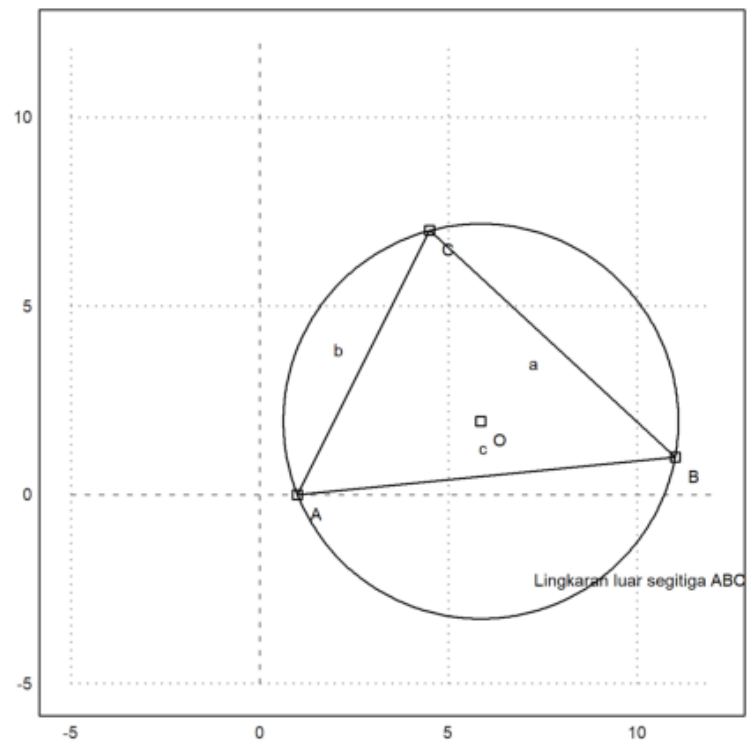
```

```

19.4560514507

```

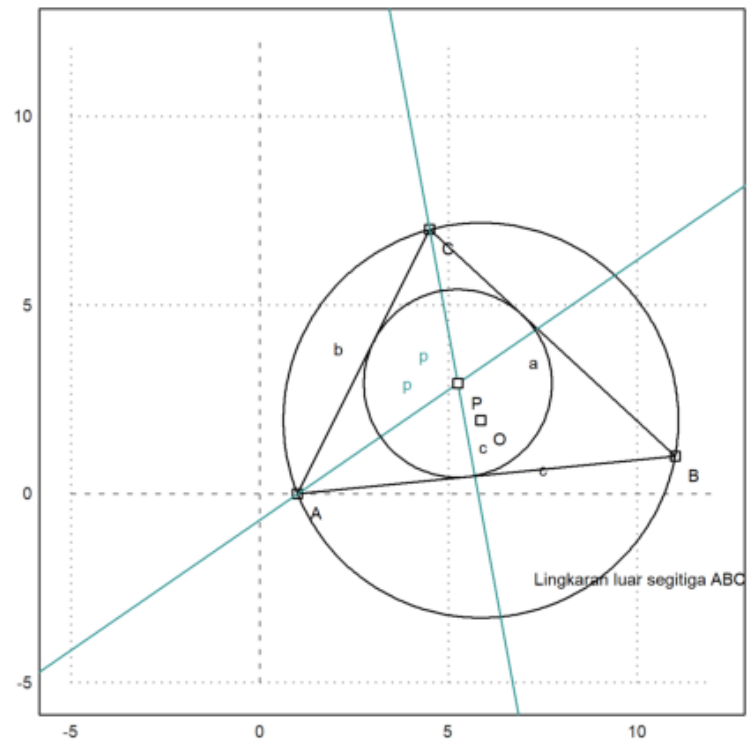
```
>plotPoint(O,"O"); // Menggambar titik "O"
>plotCircle(c,"Lingkaran luar segitiga ABC");
```



```
>l=angleBisector(A,C,B); // garis bagi <ACB
>g=angleBisector(C,A,B); // garis bagi <CAB
>i=angleBisector(A,B,C); // garis bagi <ABC
>P=lineIntersection(l,g);
>color(5); plotLine(l); plotLine(g); color(1);
>plotPoint(P,"P");
>r=norm(P-projectToLine(P,lineThrough(A,B))); // Jari-jari lingkaran dalam
>pi*r^2 // Luas lingkaran dalam
```

19.4560514507

```
>plotCircle(circleWithCenter(P,r)):
```



## Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri eksak dan simbolis menggunakan Maxima.

File geometry.e menyediakan fungsi yang sama (dan lebih) dalam Maxima. Namun, kita dapat menggunakan perhitungan simbolis sekarang.

```
>A &= [1,0]; B &= [0,1]; C &= [2,2]; // menentukan tiga titik A, B, C
```

Fungsi untuk garis dan lingkaran bekerja seperti fungsi Euler, tetapi memberikan perhitungan simbolis.

```
>c &= lineThrough(B,C) // c=BC
```

```
[- 1, 2, 2]
```

Kita dapat dengan mudah mendapatkan persamaan untuk sebuah garis.

```
>$getLineEquation(c,x,y), $solve(%,y) | expand // persamaan garis c
```

$$\left[ y = \frac{x}{2} + 1 \right]$$

$$\left[ y = \frac{x}{2} + 1 \right]$$

```
>$getLineEquation(lineThrough([x1,y1],[x2,y2]),x,y), $solve(%,y) // persamaan garis melalui
```

$$\left[ y = \frac{-(x_1 - x) y_2 - (x - x_2) y_1}{x_2 - x_1} \right]$$

$$\left[ y = \frac{-(x_1 - x) y_2 - (x - x_2) y_1}{x_2 - x_1} \right]$$

```
>$getLineEquation(lineThrough(A,[x1,y1]),x,y) // persamaan garis melalui A dan (x1, y1)
```

$$(x_1 - 1) y - x y_1 = -y_1$$

```
>h &= perpendicular(A,lineThrough(B,C)) // h melalui A tegak lurus BC
```

$$[2, 1, 2]$$

```
>Q &= lineIntersection(c,h) // Q titik potong garis c=BC dan h
```

$$\left[ -\frac{2}{5}, -\frac{6}{5} \right]$$

```
>$projectToLine(A,lineThrough(B,C)) // proyeksi A pada BC
```

$$\left[ \frac{2}{5}, \frac{6}{5} \right]$$

```
>$distance(A,Q) // jarak AQ
```

$$\frac{3}{\sqrt{5}}$$

```
>cc &= circleThrough(A,B,C); $cc // (titik pusat dan jari-jari) lingkaran melalui A, B, C
```

$$\left[ \frac{7}{6}, \frac{7}{6}, \frac{5}{3\sqrt{2}} \right]$$

```
>r:=getCircleRadius(cc); $r , $float(r) // tampilkan nilai jari-jari
```

1.178511301977579

```
>$computeAngle(A,C,B) // nilai <ACB
```

$$\arccos\left(\frac{4}{5}\right)$$

```
>$solve(getLineEquation(angleBisector(A,C,B),x,y),y)[1] // persamaan garis bagi <ACB
```

$$y = x$$

```
>P &= lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A)); $P // titik potong 2 ga
```

$$\left[ \frac{\sqrt{2}\sqrt{5}+2}{6}, \frac{\sqrt{2}\sqrt{5}+2}{6} \right]$$

```
>P() // hasilnya sama dengan perhitungan sebelumnya
```

[0.86038, 0.86038]

## Perpotongan Garis dan Lingkaran

Tentu saja, kita juga dapat memotong garis dengan lingkaran, dan lingkaran dengan lingkaran.

```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

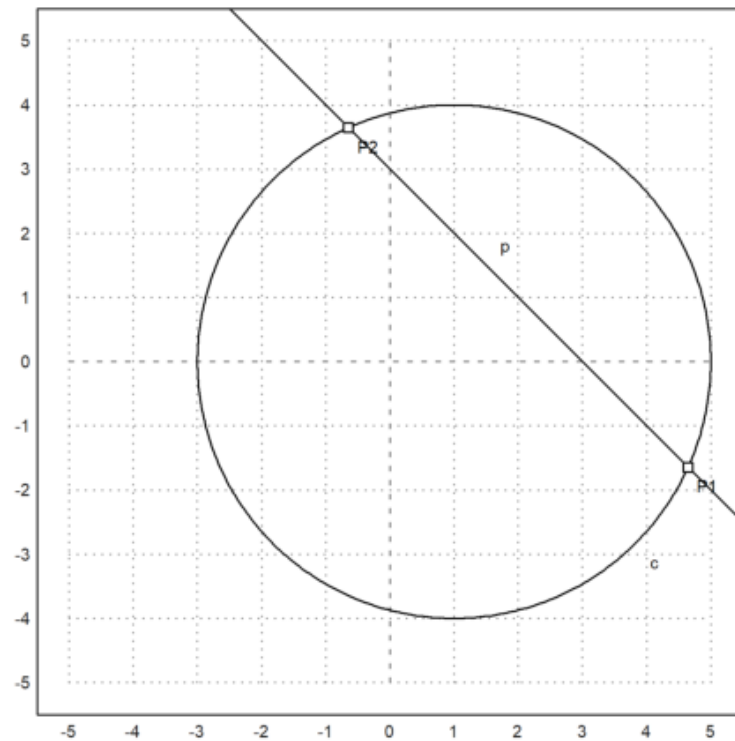
Potongan antara garis dan lingkaran menghasilkan dua titik dan jumlah titik potong.

```
>{P1,P2,f}=lineCircleIntersections(l,c);
>P1, P2, f
```



```
[4.64575, -1.64575]
[-0.645751, 3.64575]
2
```

```
>plotPoint(P1); plotPoint(P2):
```



Hal yang sama berlaku di Maxima.

```
>c := circleWithCenter(A,4) // lingkaran dengan pusat A jari-jari 4
```

```
[1, 0, 4]
```

```
>l := lineThrough(B,C) // garis l melalui B dan C
```

```
[1, 1, 3]
```

```
>$lineCircleIntersections(l,c) | radcan, // titik potong lingkaran c dan garis l
```

```
[[sqrt(7)+2, 1-sqrt(7)], [2-sqrt(7), sqrt(7)+1]]
```

Akan ditunjukkan bahwa sudut-sudut yang menghadap busur yang sama adalah sama besar.

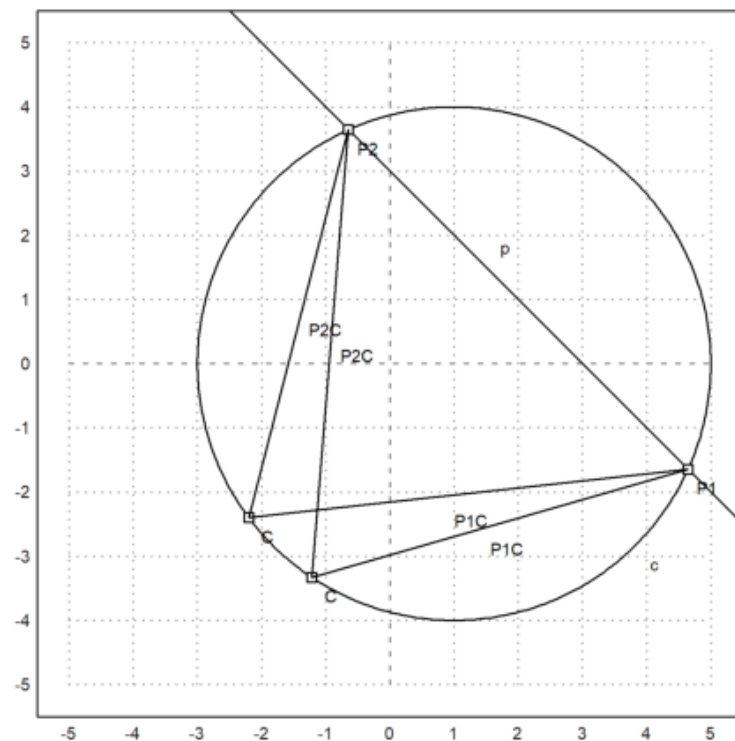
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);  
>degprint(computeAngle(P1,C,P2))
```

69°17'42.68''

```
>C=A+normalize([-4,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);  
>degprint(computeAngle(P1,C,P2))
```

69°17'42.68''

```
>insimg;
```



## Garis Sumbu

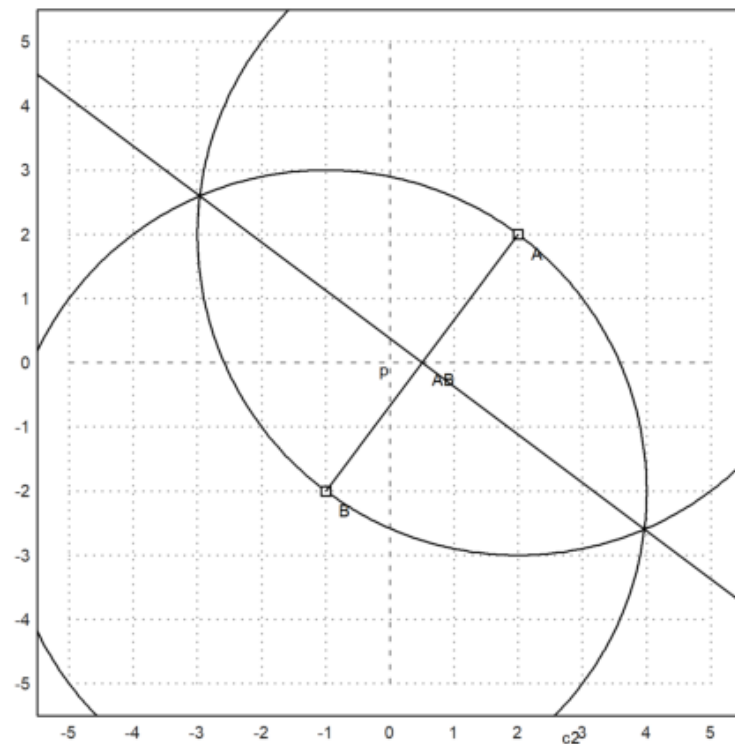
Berikut adalah langkah-langkah menggambar garis sumbu ruas garis AB:

1. Gambar lingkaran dengan pusat A melalui B.
2. Gambar lingkaran dengan pusat B melalui A.
3. Tarik garis melalui kedua titik potong kedua lingkaran tersebut. Garis ini merupakan garis sumbu (melalui titik tengah dan tegak lurus) AB.

```

>A=[2,2]; B=[-1,-2];
>c1=circleWithCenter(A,distance(A,B));
>c2=circleWithCenter(B,distance(A,B));
>{P1,P2,f}=circleCircleIntersections(c1,c2);
>l=lineThrough(P1,P2);
>setPlotRange(5); plotCircle(c1); plotCircle(c2);
>plotPoint(A); plotPoint(B); plotSegment(A,B); plotLine(l):

```



Berikutnya, kita lakukan hal yang sama di Maxima dengan koordinat umum.

```

>A &= [a1,a2]; B &= [b1,b2];
>c1 &= circleWithCenter(A,distance(A,B));
>c2 &= circleWithCenter(B,distance(A,B));
>P &= circleCircleIntersections(c1,c2); P1 &= P[1]; P2 &= P[2];

```

Persamaan untuk titik potong agak rumit. Tetapi kita bisa menyederhanakannya jika kita mencari solusi untuk y.

```

>g &= getLineEquation(lineThrough(P1,P2),x,y);
>$solve(g,y)

```

$$\left[ y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

Ini memang sama dengan garis tegak lurus tengah, yang dihitung dengan cara yang benar-benar berbeda.

```
>$solve(getLineEquation(middlePerpendicular(A,B),x,y),y)
```

$$\left[ y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

```
>h &=getLineEquation(lineThrough(A,B),x,y);
>$solve(h,y)
```

$$\left[ y = \frac{(b_2 - a_2)x - a_1b_2 + a_2b_1}{b_1 - a_1} \right]$$

Perhatikan hasil kali gradien garis g dan h adalah:

$$\frac{-(b_1 - a_1)}{(b_2 - a_2)} \times \frac{(b_2 - a_2)}{(b_1 - a_1)} = -1.$$

Artinya kedua garis tegak lurus.

### Contoh 3: Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

atau bisa ditulis dalam bentuk lain:

$$L = \frac{1}{4} \sqrt{(a+b+c)(b+c-a)(a+c-b)(a+b-c)}$$

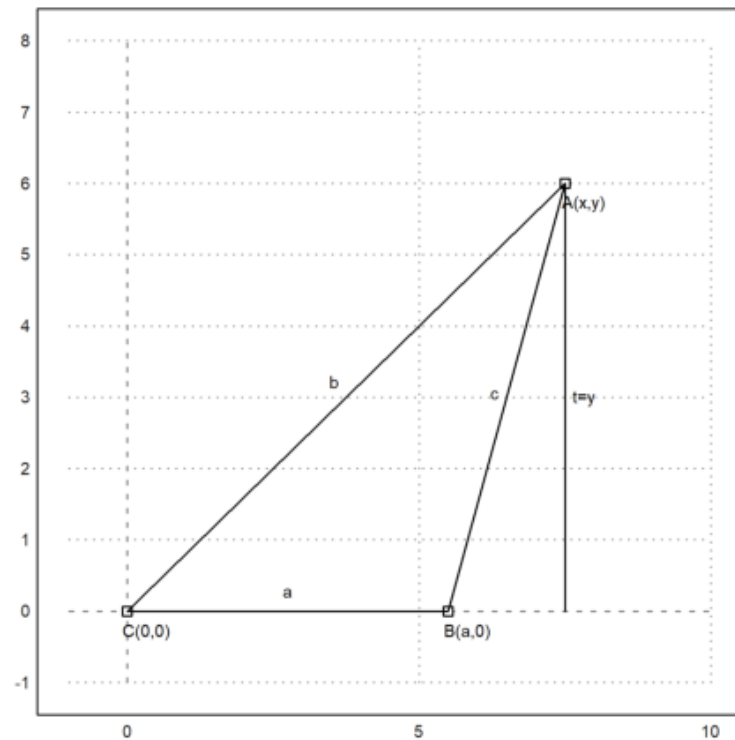
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

$$L_{\triangle ABC} = \frac{1}{2}a \times y.$$

Nilai y didapat dengan menyelesaikan sistem persamaan:

$$x^2 + y^2 = b^2, \quad (x-a)^2 + y^2 = c^2.$$

```
>setPlotRange(-1,10,-1,8); plotPoint([0,0], "C(0,0)"); plotPoint([5.5,0], "B(a,0)"; ...
> plotPoint([7.5,6], "A(x,y)");
>plotSegment([0,0],[5.5,0], "a",25); plotSegment([5.5,0],[7.5,6], "c",15); ...
>plotSegment([0,0],[7.5,6], "b",25);
>plotSegment([7.5,6],[7.5,0], "t=y",25):
```



```
>assume(a>0); sol &= solve([x^2+y^2=b^2, (x-a)^2+y^2=c^2], [x,y])
```

$$\begin{aligned} & \begin{aligned} & \begin{aligned} & -c^2 + b^2 + a^2 \\ & [x = \frac{\quad}{2a}, y = \end{aligned} \\ & \frac{\sqrt{(-c^4 + 2b^2c^2 + 2a^2c^2 - b^4 + 2ab^2 - a^4)}}{2a} \end{aligned} \\ & \begin{aligned} & -c^2 + b^2 + a^2 \\ & [x = \frac{\quad}{2a}, y = \end{aligned} \\ & \frac{\sqrt{(-c^4 + 2b^2c^2 + 2a^2c^2 - b^4 + 2ab^2 - a^4)}}{2a} \end{aligned} \end{aligned}$$

Ekstrak/selesaika untuk mendapatkan solusi y.

```
>ysol &= y with sol[2][2]; $'y=sqrt(factor(ysol^2))
```

$$y = \frac{\sqrt{(-c+b+a)(c-b+a)(c+b-a)(c+b+a)}}{2a}$$

Kita mendapatkan rumus Heron.

```
>function H(a,b,c) &= sqrt(factor((ysol*a/2)^2)); $'H(a,b,c)=H(a,b,c)
```

$$H(a,b,c) = \frac{\sqrt{(-c+b+a)(c-b+a)(c+b-a)(c+b+a)}}{4}$$

```
>$'Luas=H(2,5,6) // luas segitiga dengan panjang sisi-sisi 2, 5, 6
```

$$Luas = \frac{3\sqrt{39}}{4}$$

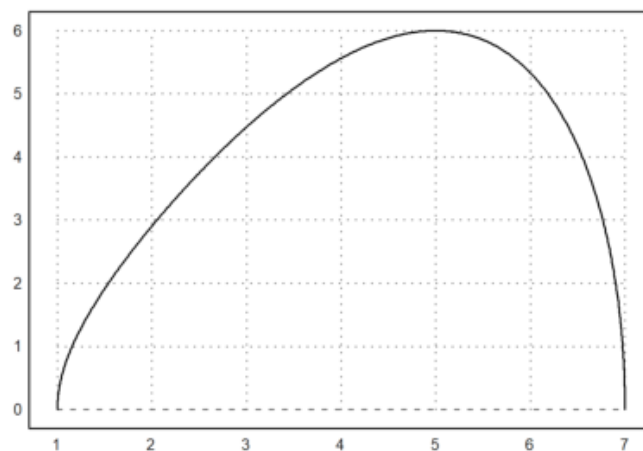
Tentu saja, setiap segitiga siku-siku adalah kasus yang sudah dikenal.

```
>H(3,4,5) //luas segitiga siku-siku dengan panjang sisi 3, 4, 5
```

6

Dan jelas, bahwa ini adalah segitiga dengan luas maksimal dan dua sisi 3 dan 4.

```
>aspect (1.5); plot2d(&H(3,4,x),1,7): // Kurva luas segitiga sengan panjang sisi 3, 4, x (
```



Kasus umum juga berlaku.

```
>$solve(diff(H(a,b,c)^2,c)=0,c)
```

$$\left[ c = -\sqrt{b^2 + a^2}, c = \sqrt{b^2 + a^2}, c = 0 \right]$$

Sekarang mari kita temukan himpunan semua titik di mana  $b+c=d$  untuk beberapa konstanta  $d$ . Sudah diketahui bahwa ini adalah suatu elips.

```
>s1 &= subst(d-c,b,sol[2]); $s1
```

$$\left[ x = \frac{(d-c)^2 - c^2 + a^2}{2a}, y = \frac{\sqrt{-(d-c)^4 + 2c^2(d-c)^2 + 2a^2(d-c)^2 - c^4 + 2a^2c^2 - a^4}}{2a} \right]$$

Dan buatlah fungsi dari ini.

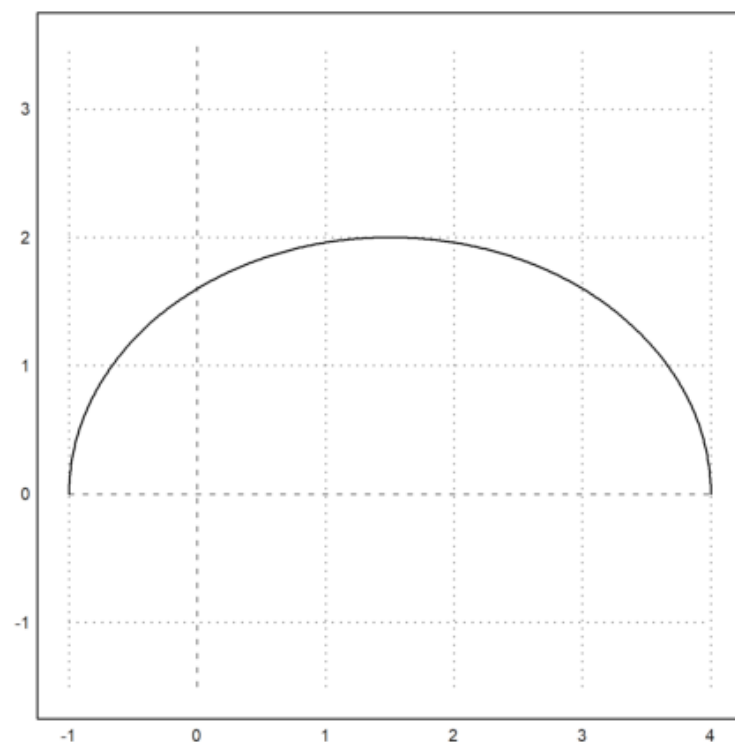
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

$$\frac{\sqrt{-(d-c)^4 + 2c^2(d-c)^2 + 2a^2(d-c)^2 - c^4 + 2a^2c^2 - a^4}}{2a}$$

$$\frac{\sqrt{-(d-c)^4 + 2c^2(d-c)^2 + 2a^2(d-c)^2 - c^4 + 2a^2c^2 - a^4}}{2a}$$

Sekarang kita bisa menggambar himpunan ini. Sisi  $b$  bervariasi dari 1 hingga 4. Sudah diketahui bahwa kita mendapatkan elips.

```
>aspect(1); plot2d(&fx(3,x,5),&fy(3,x,5),xmin=1,xmax=4,square=1):
```



Kita bisa memeriksa persamaan umum untuk elips ini, yaitu:

$$\frac{(x - x_m)^2}{u^2} + \frac{(y - y_m)^2}{v^2} = 1,$$

di mana  $(x_m, y_m)$  adalah pusat, dan  $u$  dan  $v$  adalah separuh sumbu.

```
>$ratsimp((fx(a,c,d)-a/2)^2/u^2+fy(a,c,d)^2/v^2 with [u=d/2,v=sqrt(d^2-a^2)/2])
```

1

Kita melihat bahwa tinggi dan, dengan demikian, luas segitiga maksimal untuk  $x=0$ . Jadi luas segitiga dengan  $a+b+c=d$  maksimal jika itu segitiga sama sisi. Kita ingin menurunkannya secara analitis.

```
>eqns &= [diff(H(a,b,d-(a+b))^2,a)=0,diff(H(a,b,d-(a+b))^2,b)=0]; $eqns
```

$$\left[ \frac{d(d-2a)(d-2b)}{8} - \frac{(-d+2b+2a)d(d-2b)}{8} = 0, \frac{d(d-2a)(d-2b)}{8} - \frac{(-d+2b+2a)d(d-2a)}{8} = 0 \right]$$

Kita mendapatkan beberapa nilai minimum, yang berkaitan dengan segitiga dengan satu sisi 0, dan solusi  $a=b=c=d/3$ .

```
>$solve(eqns,[a,b])
```

$$\left[ \left[ a = \frac{d}{3}, b = \frac{d}{3} \right], \left[ a = 0, b = \frac{d}{2} \right], \left[ a = \frac{d}{2}, b = 0 \right], \left[ a = \frac{d}{2}, b = \frac{d}{2} \right] \right]$$

Ada juga metode Lagrange, memaksimalkan  $H(a,b,c)^2$  dengan memperhatikan  $a+b+c=d$ .

```
>&solve([diff(H(a,b,c)^2,a)=la,diff(H(a,b,c)^2,b)=la, ...  
> diff(H(a,b,c)^2,c)=la,a+b+c=d],[a,b,c,la])
```

$$\begin{aligned} & \left[ \left[ a = 0, b = \frac{d}{2}, c = \frac{d}{2}, la = 0 \right], \right. \\ & \left[ a = \frac{d}{2}, b = 0, c = \frac{d}{2}, la = 0 \right], \left[ a = \frac{d}{2}, b = \frac{d}{2}, c = 0, la = 0 \right], \\ & \left. \left[ a = \frac{d}{3}, b = \frac{d}{3}, c = \frac{d}{3}, la = \frac{d}{108} \right] \right] \end{aligned}$$

Kita bisa membuat plot situasi ini.



Pertama-tama, tentukan titik-titiknya di Maxima.

```
>A &= at([x,y],sol[2]); $A
```

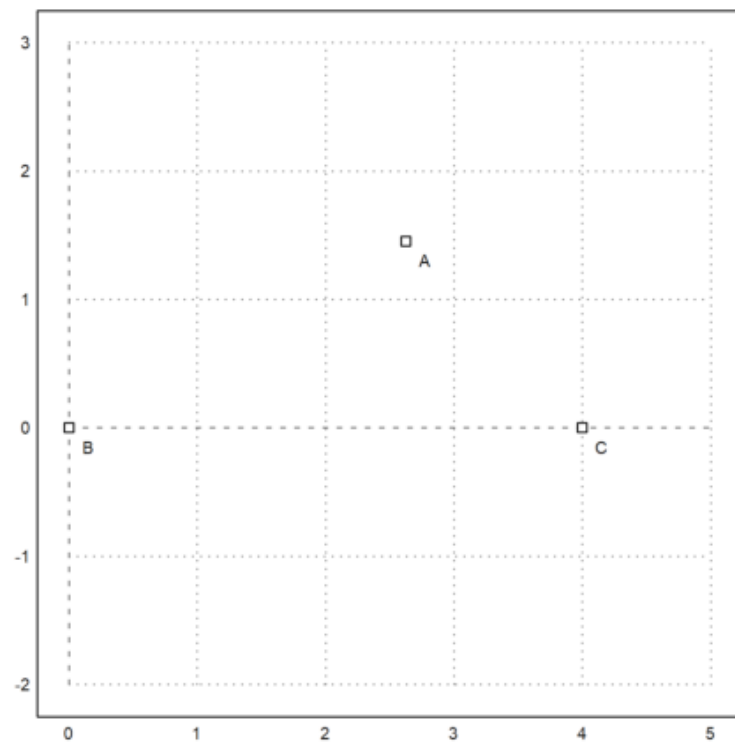
$$\left[ \frac{-c^2 + b^2 + a^2}{2a}, \frac{\sqrt{-c^4 + 2b^2c^2 + 2a^2c^2 - b^4 + 2a^2b^2 - a^4}}{2a} \right]$$

```
>B &= [0,0]; $B, C &= [a,0]; $C
```

$$[a, 0]$$

Kemudian atur rentang plot, dan gambar plot titik-titik tersebut.

```
>setPlotRange(0,5,-2,3); ...
>a=4; b=3; c=2; ...
>plotPoint(mxmeval("B"),"B"); plotPoint(mxmeval("C"),"C"); ...
>plotPoint(mxmeval("A"),"A");
```

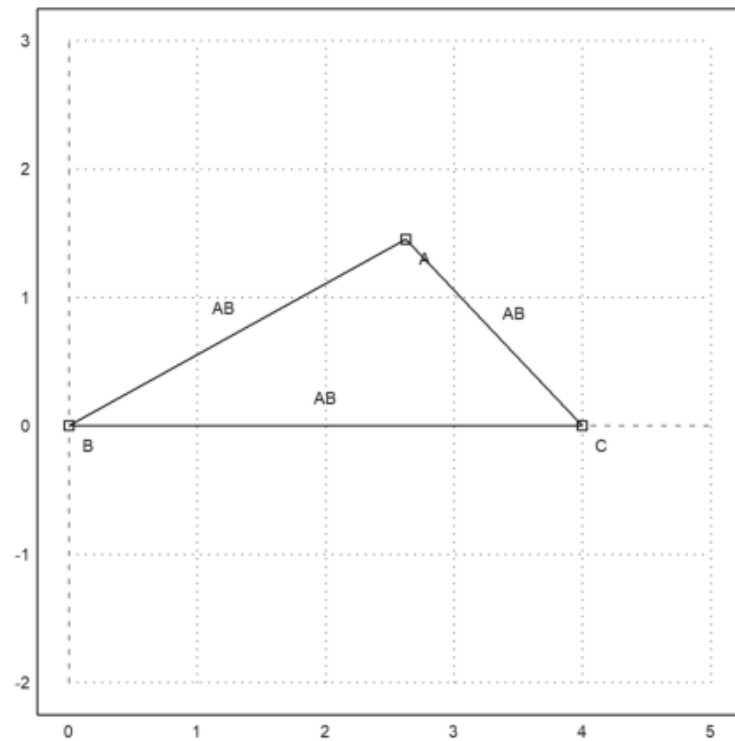


Buat plot segmen-segmen tersebut.

```

>plotSegment(mxmeval("A"),mxmeval("C")); ...
>plotSegment(mxmeval("B"),mxmeval("C")); ...
>plotSegment(mxmeval("B"),mxmeval("A")):

```



Hitung garis tegak lurus tengah di Maxima.

```

>h &= middlePerpendicular(A,B); g &= middlePerpendicular(B,C);

```

Dan pusat lingkaran.

```

>U &= lineIntersection(h,g);

```

Kita mendapatkan rumus untuk jari-jari lingkaran.

```

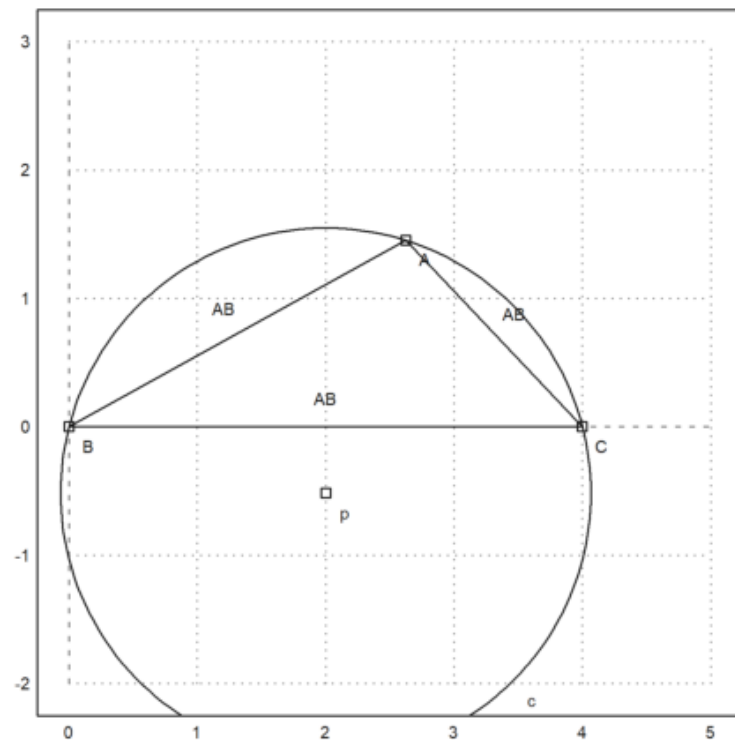
>&assume(a>0,b>0,c>0); $distance(U,B) | radcan

```

$$\frac{abc}{\sqrt{c-b-a}\sqrt{c-b+a}\sqrt{c+b-a}\sqrt{c+b+a}}$$

Mari tambahkan ini ke plot.

```
>plotPoint(U()); ...
>plotCircle(circleWithCenter(mxmeval("U"),mxmeval("distance(U,C)"))):
```



Dengan menggunakan geometri, kita turunkan rumus sederhana

$$\frac{a}{\sin(\alpha)} = 2r$$

untuk jari-jari. Kita bisa memeriksa apakah ini benar-benar benar dengan Maxima. Maxima akan memfaktorkan ini hanya jika kita mengkuadratkannya.

```
>$c^2/sin(computeAngle(A,B,C))^2 | factor
```

$$-\frac{4a^2b^2c^2}{(c-b-a)(c-b+a)(c+b-a)(c+b+a)}$$

## Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari segitiga apa pun yang tidak sama sisi. Ini adalah garis tengah segitiga, dan melalui beberapa titik penting yang ditentukan dari segitiga tersebut, termasuk ortosenter, pusat lingkaran luar, centroid, titik Exeter, dan pusat lingkaran sembilan titik segitiga.

Sebagai demonstrasi, kita menghitung dan memplot garis Euler dalam suatu segitiga.

Pertama, kita tentukan sudut-sudut segitiga dalam Euler. Kita menggunakan definisi yang terlihat dalam ekspresi simbolis.

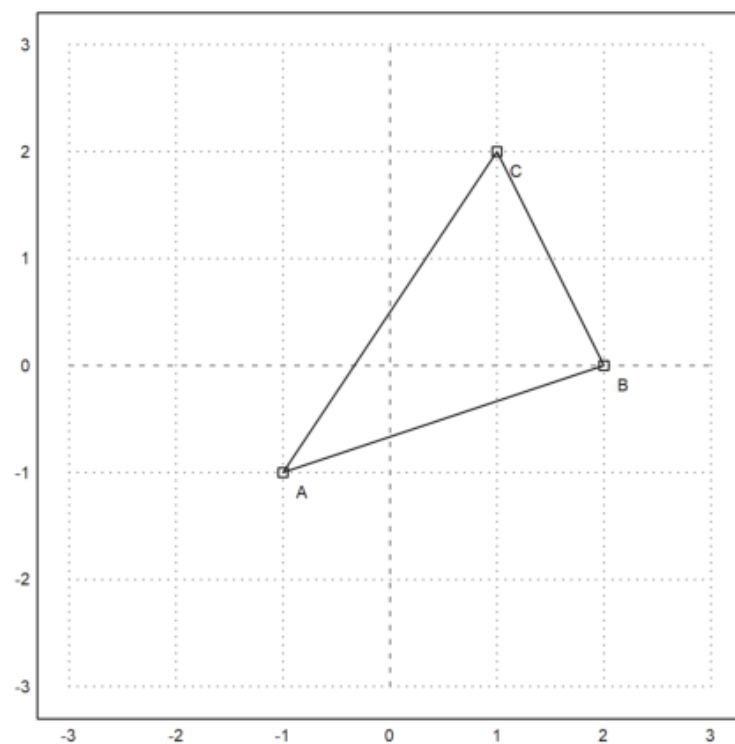
```
>A:=[-1,-1]; B:=[2,0]; C:=[1,2];
```

Untuk memplot objek-objek geometris, kita siapkan area plot, dan tambahkan titik-titik ke dalamnya. Semua plot objek geometris ditambahkan ke plot saat ini.

```
>setPlotRange(3); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C");
```

Kita juga dapat menambahkan sisi-sisi segitiga.

```
>plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,A,""):
```



Berikut adalah luas segitiga, menggunakan rumus determinan. Tentu saja, kita harus mengambil nilai absolut dari hasil ini.

```
>$areaTriangle(A,B,C)
```

$$-\frac{7}{2}$$

Kita dapat menghitung koefisien dari sisi c.

```
>c &= lineThrough(A,B)
```

$$[-1, 3, -2]$$

Dan juga mendapatkan rumus untuk garis ini.

```
>$getLineEquation(c,x,y)
```

$$3y - x = -2$$

Untuk bentuk Hesse, kita perlu menentukan suatu titik, sehingga titik tersebut berada di sisi positif bentuk Hesse. Memasukkan titik menghasilkan jarak positif ke garis.

```
>$getHesseForm(c,x,y,C), $at(%, [x=C[1],y=C[2]])
```

$$\frac{7}{\sqrt{10}}$$

$$\frac{7}{\sqrt{10}}$$

Sekarang kita menghitung lingkaran luar ABC.

```
>LL &= circleThrough(A,B,C); $getCircleEquation(LL,x,y)
```

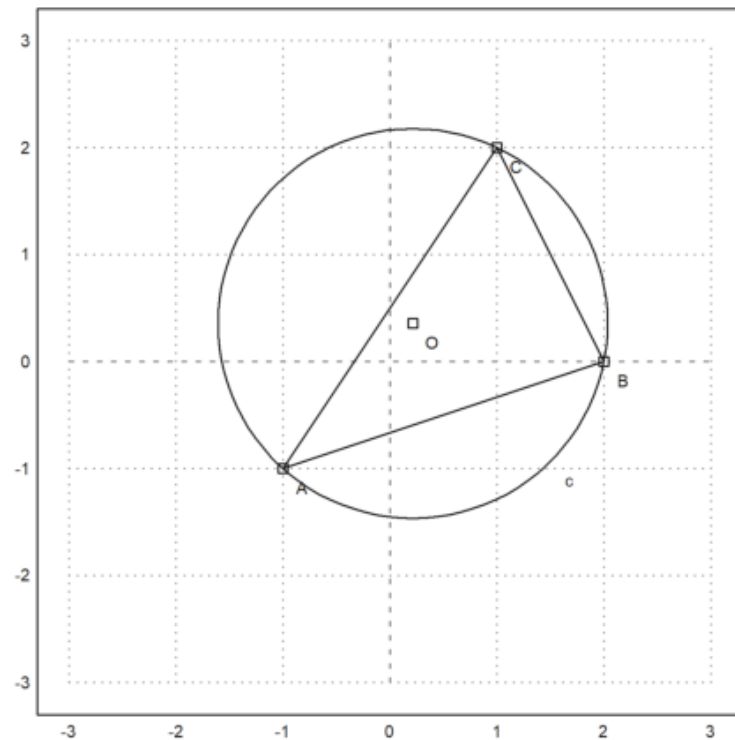
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

```
>O &= getCircleCenter(LL); $O
```

$$\left[\frac{3}{14}, \frac{5}{14}\right]$$

Plot lingkaran dan pusatnya. Cu dan U bersifat simbolis. Kita mengevaluasi ekspresi ini untuk Euler.

```
>plotCircle(LL()); plotPoint(O(),"O"):
```



Kita dapat menghitung irisan dari tinggi segitiga ABC (ortosenter) secara numerik dengan perintah berikut.

```
>H <- lineIntersection(perpendicular(A,lineThrough(C,B)),...
> perpendicular(B,lineThrough(A,C))); $H
```

$$\left[ \frac{11}{7}, \frac{2}{7} \right]$$

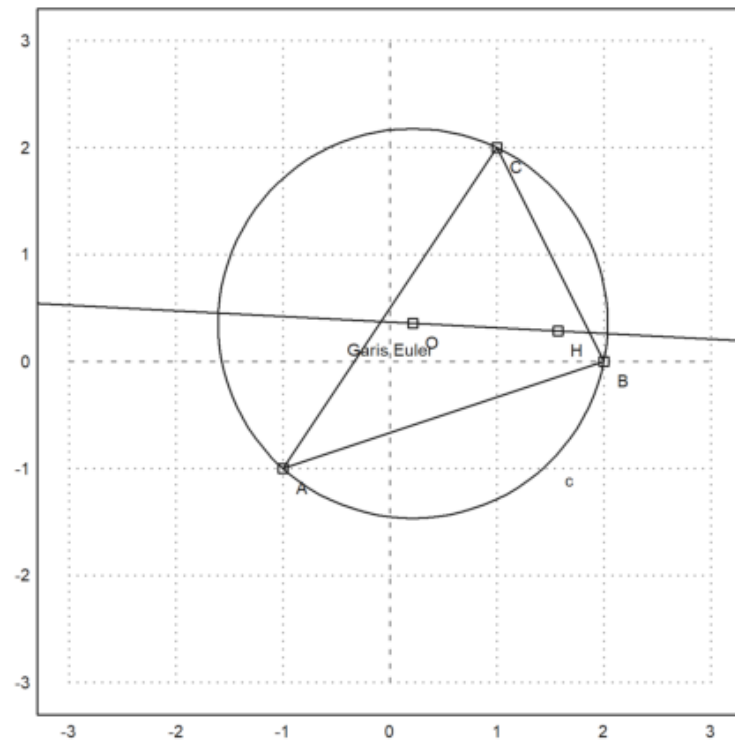
Sekarang kita dapat menghitung garis Euler dari segitiga.

```
>el <- lineThrough(H,O); $getLineEquation(el,x,y)
```

$$-\frac{19y}{14} - \frac{x}{14} = -\frac{1}{2}$$

Tambahkan ke plot yang sudah dibuat.

```
>plotPoint(H(),"H"); plotLine(el(),"Garis Euler"):
```

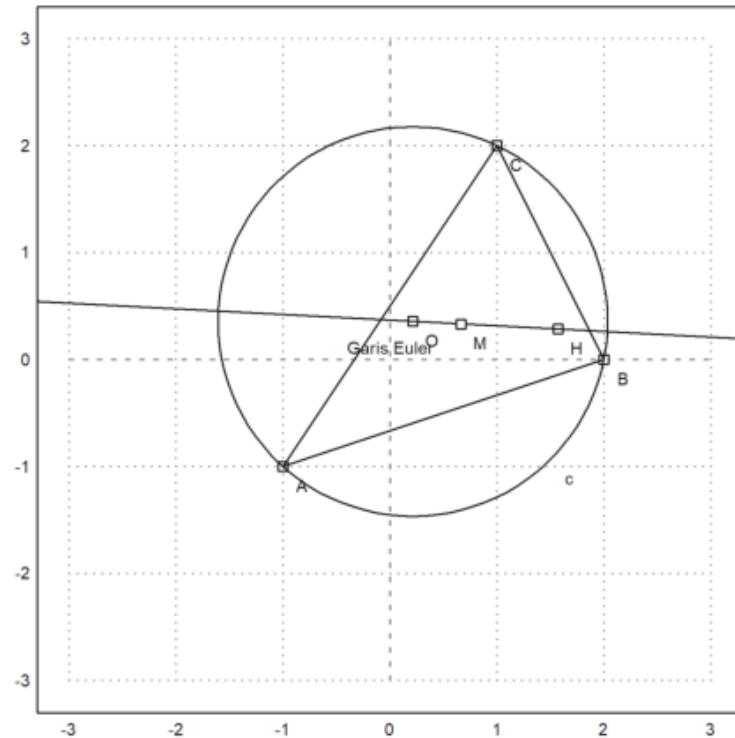


Pusat gravitasi seharusnya berada pada garis ini.

```
>M &= (A+B+C)/3; $getLineEquation(el,x,y) with [x=M[1],y=M[2]]
```

$$-\frac{1}{2} = -\frac{1}{2}$$

```
>plotPoint(M(),"M"): // titik berat
```



Teori memberi tahu kita  $MH=2*MO$ . Kita perlu menyederhanakan dengan radcan untuk mencapai ini.

```
>$distance(M,H)/distance(M,O)|radcan
```

2

Fungsi-fungsi mencakup fungsi untuk sudut juga.

```
>$computeAngle(A,C,B), degprint(%())
```

$$\arccos\left(\frac{4}{\sqrt{5}\sqrt{13}}\right)$$

$60^{\circ}15'18.43''$

Persamaan untuk pusat lingkaran dalam adalah tidak begitu bagus.

```
>Q &= lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A))|radcan; $Q
```

$$\left[ \frac{\left(2^{\frac{3}{2}} + 1\right) \sqrt{5} \sqrt{13} - 15 \sqrt{2} + 3}{14}, \frac{(\sqrt{2} - 3) \sqrt{5} \sqrt{13} + 5 \cdot 2^{\frac{3}{2}} + 5}{14} \right]$$



Mari kita juga menghitung ungkapan untuk jari-jari lingkaran dalam.

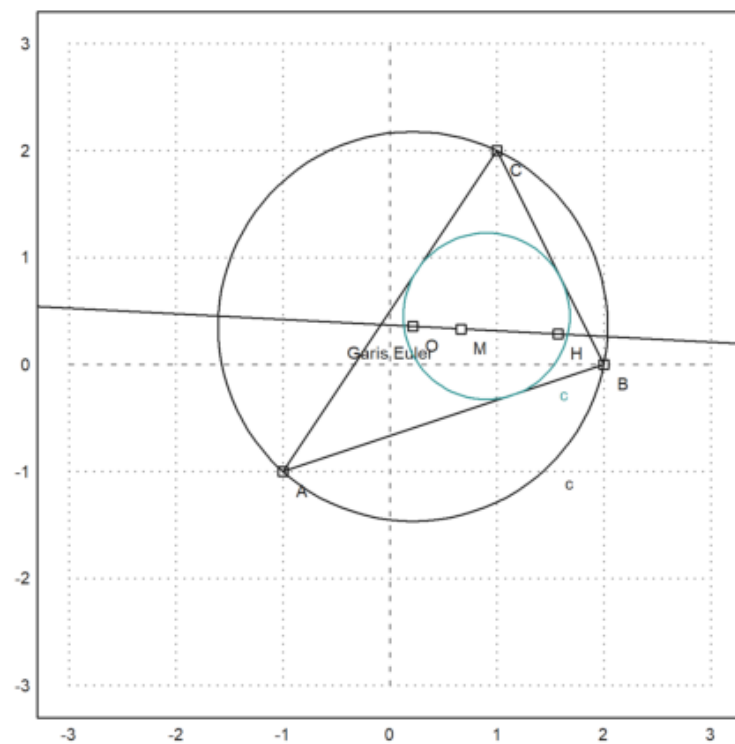
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))/ratsimp; $r
```

$$\frac{\sqrt{(-41\sqrt{2}-31)\sqrt{5}\sqrt{13}+115\sqrt{2}+614}}{7\sqrt{2}}$$

```
>LD &= circleWithCenter(Q,r); // Lingkaran dalam
```

Mari tambahkan ini ke dalam plot.

```
>color(5); plotCircle(LD());
```



## Parabola

Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

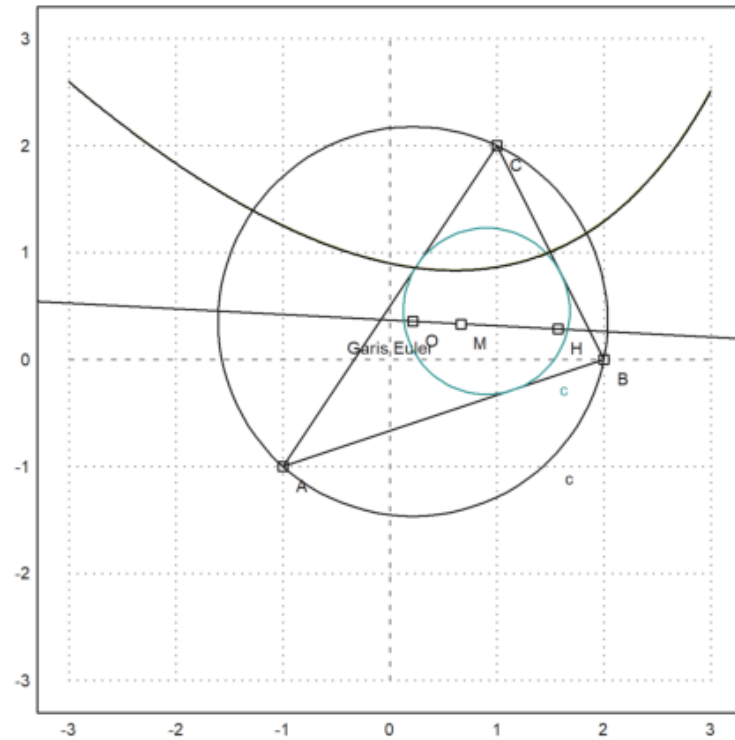
```
>p &= getHesseForm(lineThrough(A,B),x,y,C)-distance([x,y],C); $p='0
```

$$\frac{3y-x+2}{\sqrt{10}} - \sqrt{(2-y)^2 + (1-x)^2} = 0$$

```
>plot2d(p, level=0, add=1, contourcolor=6):
```



```
>plot2d(&rhs(akar[1]), add=1):
```



```
>function g(x) &= rhs(akar[1]); $'g(x)= g(x) // fungsi yang mendefinisikan kurva di atas
```

$$g(x) = -3x - \sqrt{70}\sqrt{9-2x} + 26$$

```
>T &= [-1, g(-1)]; // ambil sebarang titik pada kurva tersebut
>dTC &= distance(T,C); $fullratsimp(dTC), $float(%) // jarak T ke C
```

2.135605779339061

```
>U &= projectToLine(T,lineThrough(A,B)); $U // proyeksi T pada garis AB
```

$$\left[ \frac{80 - 3\sqrt{11}\sqrt{70}}{10}, \frac{20 - \sqrt{11}\sqrt{70}}{10} \right]$$

```
>dU2AB &= distance(T,U); $fullratsimp(dU2AB), $float(%) // jarak T ke AB
```

2.135605779339061

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

## Contoh 5: Trigonometri Rasional

Ini terinspirasi dari pembicaraan N.J.Wildberger. Dalam bukunya "Divine Proportions", Wildberger mengusulkan untuk menggantikan konsep klasik jarak dan sudut dengan kuadrans dan spread. Dengan menggunakan ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh dan tetap "rasional".

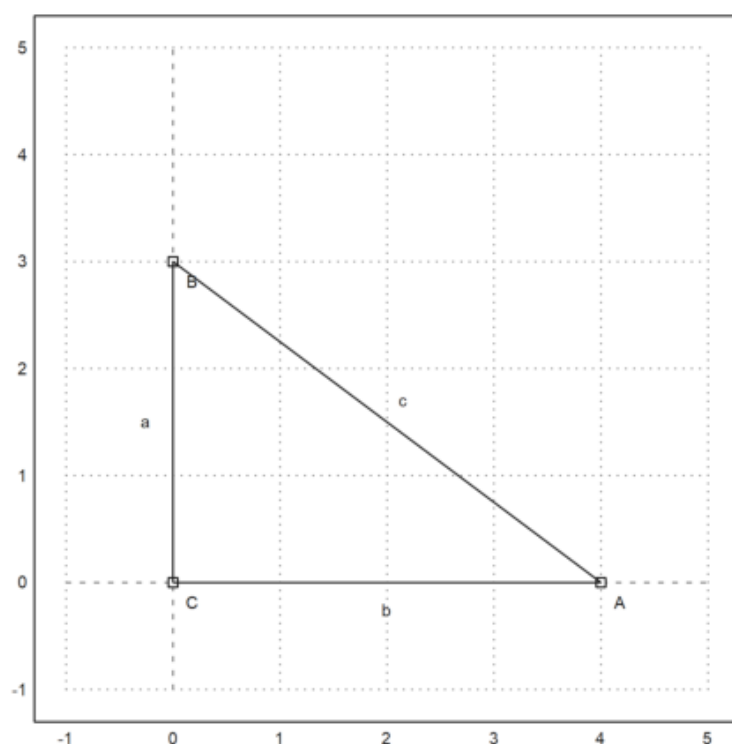
Berikut, saya memperkenalkan konsep-konsep tersebut dan memecahkan beberapa masalah. Saya menggunakan komputasi simbolik Maxima di sini, yang menyembunyikan keuntungan utama trigonometri rasional bahwa komputasi dapat dilakukan hanya dengan kertas dan pensil. Anda diundang untuk memeriksa hasil tanpa komputer.

Intinya adalah bahwa komputasi simbolik rasional sering memberikan hasil yang sederhana. Sebaliknya, trigonometri klasik memberikan hasil trigonometri yang rumit, yang hanya dievaluasi menjadi perkiraan numerik.

```
>load geometry;
```

Untuk pengenalan pertama, kita menggunakan segitiga siku-siku dengan proporsi Mesir terkenal 3, 4, dan 5. Perintah-perintah berikut adalah perintah Euler untuk merencanakan geometri datar yang terkandung dalam file Euler "geometry.e".

```
>C&:=[0,0]; A&:=[4,0]; B&:=[0,3]; ...
>setPlotRange(-1,5,-1,5); ...
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
>plotSegment(B,A,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...
>insimg(30);
```



Tentu saja,

$$\sin(w_a) = \frac{a}{c},$$

di mana  $w_a$  adalah sudut di A. Cara biasa untuk menghitung sudut ini adalah dengan mengambil invers dari fungsi sinus. Hasilnya adalah sudut yang sulit dicerna, yang hanya bisa dicetak dengan perkiraan numerik.

```
>wa := arcsin(3/5); degprint(wa)
```

36°52'11.63''

Trigonometri rasional berusaha menghindari ini.

Notasi pertama trigonometri rasional adalah kuadrans, yang menggantikan jarak. Sebenarnya, itu hanyalah jarak yang dikuadratkan. Berikutnya,  $a$ ,  $b$ , dan  $c$  menyatakan kuadrans dari sisi-sisi.

Teorema Pythagoras dengan mudah menjadi  $a+b=c$  kemudian.

```
>a &= 3^2; b &= 4^2; c &= 5^2; &a+b=c
```

$$25 = 25$$

Notasi kedua trigonometri rasional adalah spread. Spread mengukur pembukaan antara garis. Nilainya 0 jika garisnya sejajar, dan 1 jika garisnya tegak lurus. Ini adalah kuadrat dari sinus sudut antara dua garis.

Spread dari garis AB dan AC dalam gambar di atas didefinisikan sebagai

$$s_a = \sin(\alpha)^2 = \frac{a}{c},$$

di mana  $a$  dan  $c$  adalah kuadrans dari segitiga siku-siku dengan satu sudut di A.

```
>sa &= a/c; $sa
```

$$\frac{9}{25}$$

Ini lebih mudah dihitung daripada sudut, tentu saja. Tetapi Anda kehilangan properti bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita bisa mengonversi nilai perkiraan kita untuk sudut  $w_a$  menjadi spread dan mencetaknya sebagai pecahan.

```
>fracprint(sin(wa)^2)
```

9/25

Hukum kosinus trigonometri klasik diterjemahkan menjadi "hukum silang" berikut.

$$(c + b - a)^2 = 4bc(1 - s_a)$$

Di sini  $a$ ,  $b$ , dan  $c$  adalah kuadrans dari sisi-sisi segitiga, dan  $sa$  adalah spread di sudut  $A$ . Sisi  $a$ , seperti biasa, berlawanan dengan sudut  $A$ .

Hukum-hukum ini diimplementasikan dalam file `geometry.e` yang kita muat ke dalam Euler.

```
>$crosslaw(aa,bb,cc,saa)
```

$$(cc + bb - aa)^2 = 4 bb cc (1 - saa)$$

Dalam kasus ini, kita mendapatkan

```
>$crosslaw(a,b,c,sa)
```

$$1024 = 1024$$

Mari kita gunakan hukum silang ini untuk menemukan spread di  $A$ . Untuk melakukannya, kita menghasilkan hukum silang untuk kuadrans  $a$ ,  $b$ , dan  $c$ , dan memecahkannya untuk spread  $sa$  yang tidak diketahui.

Anda dapat melakukannya dengan mudah secara manual, tetapi saya menggunakan Maxima. Tentu saja, kita mendapatkan hasil yang sudah kita miliki.

```
>$crosslaw(a,b,c,x), $solve(%,x)
```

$$\left[ x = \frac{9}{25} \right]$$

$$\left[ x = \frac{9}{25} \right]$$

Ini sudah kita ketahui. Definisi spread adalah kasus khusus dari hukum silang.

Kita juga bisa memecahkan ini untuk  $a$ ,  $b$ ,  $c$  umum. Hasilnya adalah formula yang menghitung spread sudut segitiga yang diberikan kuadrans dari tiga sisi.

```
>$solve(crosslaw(aa,bb,cc,x),x)
```

$$\left[ x = \frac{-cc^2 - (-2bb - 2aa)cc - bb^2 + 2aa bb - aa^2}{4bbcc} \right]$$

Kita bisa membuat fungsi dari hasil ini. Fungsi tersebut sudah didefinisikan dalam file `geometry.e` dari Euler.

```
>$spread(a,b,c)
```

$$\frac{9}{25}$$

Sebagai contoh, kita bisa menggunakannya untuk menghitung sudut segitiga dengan sisi-sisi

$$a, \quad a, \quad \frac{4a}{7}$$

Hasilnya rasional, yang tidak semudah itu jika kita menggunakan trigonometri klasik.

```
>$spread(a,a,4*a/7)
```

$$\frac{6}{7}$$

Ini adalah sudut dalam derajat.

```
>degprint(arcsin(sqrt(6/7)))
```

67°47'32.44''

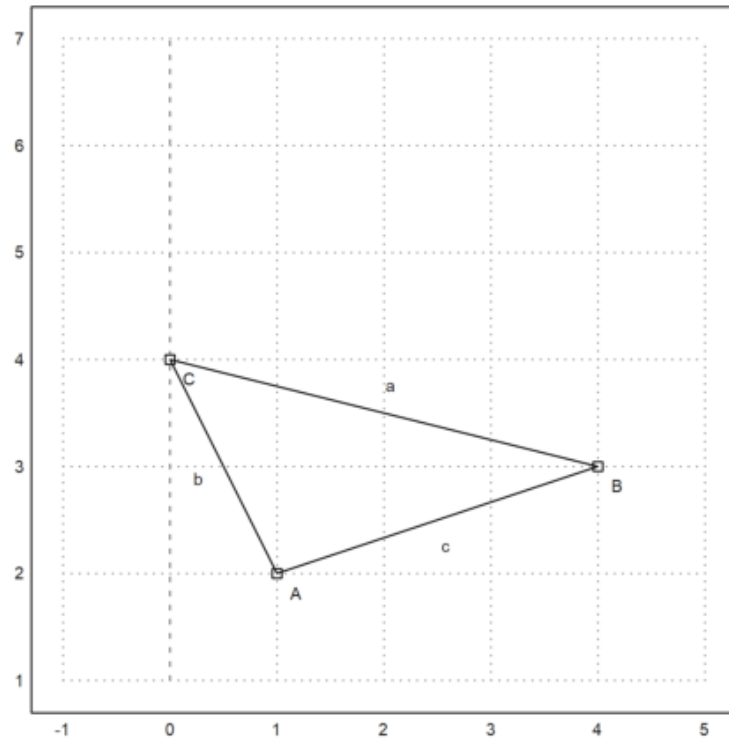
---

## Contoh Lainnya

Sekarang, mari kita coba contoh yang lebih advanced/canggih.

Kita menetapkan tiga sudut dari sebuah segitiga seperti berikut.

```
>A&:=[1,2]; B&:=[4,3]; C&:=[0,4]; ...  
>setPlotRange(-1,5,1,7); ...  
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...  
>plotSegment(B,A,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...  
>insimg;
```



Menggunakan Pythagoras, mudah untuk menghitung jarak antara dua titik. Saya pertama kali menggunakan fungsi jarak dari file Euler untuk geometri. Fungsi jarak menggunakan geometri klasik.

```
>$distance(A,B)
```

$$\sqrt{10}$$

Euler juga memiliki fungsi untuk kuadrans antara dua titik.

Pada contoh berikut, karena  $c+b$  bukan  $a$ , segitiga tersebut tidak bersudut kanan.

```
>c &= quad(A,B); $c, b &= quad(A,C); $b, a &= quad(B,C); $a,
```

17

Pertama, mari kita hitung sudut tradisional. Fungsi `computeAngle` menggunakan metode biasa berdasarkan hasil kali dot dari dua vektor. Hasilnya adalah perkiraan titik ambang desimal.

$$A = \langle 1, 2 \rangle \quad B = \langle 4, 3 \rangle, \quad C = \langle 0, 4 \rangle$$

$$\mathbf{a} = C - B = \langle -4, 1 \rangle, \quad \mathbf{c} = A - B = \langle -3, -1 \rangle, \quad \beta = \angle ABC$$



$$\mathbf{a} \cdot \mathbf{c} = |\mathbf{a}| \cdot |\mathbf{c}| \cos \beta$$

$$\cos \angle ABC = \cos \beta = \frac{\mathbf{a} \cdot \mathbf{c}}{|\mathbf{a}| \cdot |\mathbf{c}|} = \frac{12 - 1}{\sqrt{17}\sqrt{10}} = \frac{11}{\sqrt{17}\sqrt{10}}$$

```
>wb &= computeAngle(A,B,C); $wb, $(wb/pi*180)()
```

$$\arccos\left(\frac{11}{\sqrt{10}\sqrt{17}}\right)$$

32.4711922908

Dengan menggunakan pensil dan kertas, kita bisa melakukan hal yang sama dengan hukum silang. Kita masukkan kuadrans a, b, dan c ke dalam hukum silang dan selesaikan untuk x.

```
>$crosslaw(a,b,c,x), $solve(%,x), //(b+c-a)^=4b.c(1-x)
```

$$\left[ x = \frac{49}{50} \right]$$

$$\left[ x = \frac{49}{50} \right]$$

Itulah yang dilakukan oleh fungsi spread yang didefinisikan dalam "geometry.e".

```
>sb &= spread(b,a,c); $sb
```

$$\frac{49}{170}$$

Maxima mendapatkan hasil yang sama menggunakan trigonometri biasa, jika kita memaksa. Ia mengurai istilah  $\sin(\arccos(...))$  menjadi hasil pecahan. Sebagian besar siswa mungkin tidak bisa melakukannya.

```
>$sin(computeAngle(A,B,C))^2
```

$$\frac{49}{170}$$

Setelah kita memiliki ambang di B, kita dapat menghitung tinggi  $h_a$  di sisi a. Ingat bahwa

$$s_b = \frac{h_a}{c}$$

menurut definisi.

```
>ha &= c*sb; $ha
```

$$\frac{49}{17}$$

Gambar berikut ini telah dihasilkan dengan program geometri C.a.R., yang dapat menggambar kuadrans dan spread.

image: (20) Rational\_Geometry\_CaR.png

Menurut definisi, panjang ha adalah akar kuadrat kuadransnya.

```
>$sqrt (ha)
```

$$\frac{7}{\sqrt{17}}$$

Sekarang kita bisa menghitung luas segitiga. Jangan lupa, bahwa kita berurusan dengan kuadrans!

```
>$sqrt (ha) *sqrt (a) /2
```

$$\frac{7}{2}$$

Formula determinan biasa memberikan hasil yang sama.

```
>$areaTriangle (B,A,C)
```

$$\frac{7}{2}$$

## Rumus Heron

---

Sekarang, mari kita selesaikan masalah ini secara umum!

```
>&remvalue (a,b,c, sb, ha) ;
```

Pertama, kita hitung ambang di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita hitung luasnya dikuadratkan (mungkin "quadrea"?), faktorkan dengan Maxima, dan kita dapatkan rumus terkenal Heron.

Mengakui, ini sulit dilakukan dengan pensil dan kertas.

```
>$spread (b^2,c^2,a^2) , $factor (%*c^2*a^2/4)
```

$$\frac{(-c+b+a)(c-b+a)(c+b-a)(c+b+a)}{16}$$

$$\frac{(-c+b+a)(c-b+a)(c+b-a)(c+b+a)}{16}$$

## Aturan Triple Spread

---

Kerugiannya adalah bahwa mereka tidak lagi hanya menambah seperti sudut. Namun, tiga ambang dari sebuah segitiga memenuhi aturan "triple spread" berikut.

```
>remvalue(sa,sb,sc); $triplespread(sa,sb,sc)
```

$$(sc + sb + sa)^2 = 2 (sc^2 + sb^2 + sa^2) + 4 sa sb sc$$

Aturan ini berlaku untuk tiga sudut apa pun yang ditambahkan menjadi  $180^\circ$ .

$$\alpha + \beta + \gamma = \pi$$

Karena ambang dari

$$\alpha, \pi - \alpha$$

sama, aturan tiga ambang juga benar, jika

$$\alpha + \beta = \gamma$$

Karena ambang dari sudut negatif adalah sama, aturan tiga ambang juga berlaku, jika

$$\alpha + \beta + \gamma = 0$$

Sebagai contoh, kita dapat menghitung ambang dari sudut  $60^\circ$ . Itu adalah  $3/4$ . Persamaan tersebut memiliki solusi kedua, di mana semua ambang adalah 0.

```
>$solve(triplespread(x,x,x),x)
```

$$\left[ x = \frac{3}{4}, x = 0 \right]$$

Ambang dari  $90^\circ$  jelas 1. Jika dua sudut ditambahkan menjadi  $90^\circ$ , ambang mereka memenuhi persamaan tiga ambang dengan a,b,1. Dengan perhitungan berikutnya, kita mendapatkan a+b=1.

```
>$triplespread(x,y,1), $solve(%,x)
```

$$[x = 1 - y]$$

Karena ambang dari  $180^\circ - t$  sama dengan ambang dari t, rumus tiga ambang juga berlaku, jika satu sudut adalah jumlah atau selisih dari dua sudut lainnya.

Jadi kita dapat menemukan ambang dari sudut yang digandakan. Perhatikan bahwa ada dua solusi lagi. Kita membuat ini menjadi fungsi.

```
>$solve(triplespread(a,a,x),x), function doublespread(a) &= factor(rhs(%[1]))
```

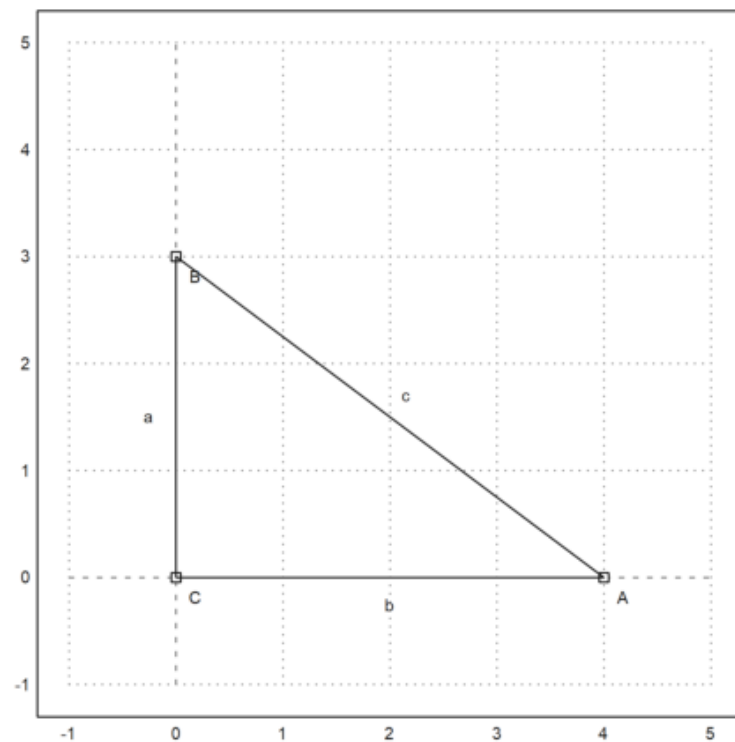
$$[x = 4a - 4a^2, x = 0]$$

$$-4(a-1)a$$

## Sudut Bisectors/Pembagi

Inilah situasinya yang sudah kita ketahui.

```
>C&:=[0,0]; A&:=[4,0]; B&:=[0,3]; ...
>setPlotRange(-1,5,-1,5); ...
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
>plotSegment(B,A,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...
>insimg;
```



Mari kita hitung panjang pembagi sudut di A. Tapi kita ingin menyelesaikannya untuk umum a,b,c.

```
>&remvalue(a,b,c);
```

Jadi kita pertama-tama menghitung ambang dari sudut yang dibagi di A, menggunakan aturan tiga ambang. Masalah dengan rumus ini muncul lagi. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya merujuk pada sudut yang dibagi  $180^\circ$ -wa.

```
>$triplespread(x,x,a/(a+b)), $solve(%,x), sa2 &= rhs(%[1]); $sa2
```

$$\frac{-\sqrt{b}\sqrt{b+a}+b+a}{2b+2a}$$

$$\left[ x = \frac{-\sqrt{b}\sqrt{b+a}+b+a}{2b+2a}, x = \frac{\sqrt{b}\sqrt{b+a}+b+a}{2b+2a} \right]$$

$$\frac{-\sqrt{b}\sqrt{b+a}+b+a}{2b+2a}$$

Mari kita periksa untuk persegi Mesir.

```
>$sa2 with [a=3^2,b=4^2]
```

$$\frac{1}{10}$$

Kita bisa mencetak sudut dalam Euler, setelah mentransfer ambangnya ke radian.

```
>wa2 := arcsin(sqrt(1/10)); degprint(wa2)
```

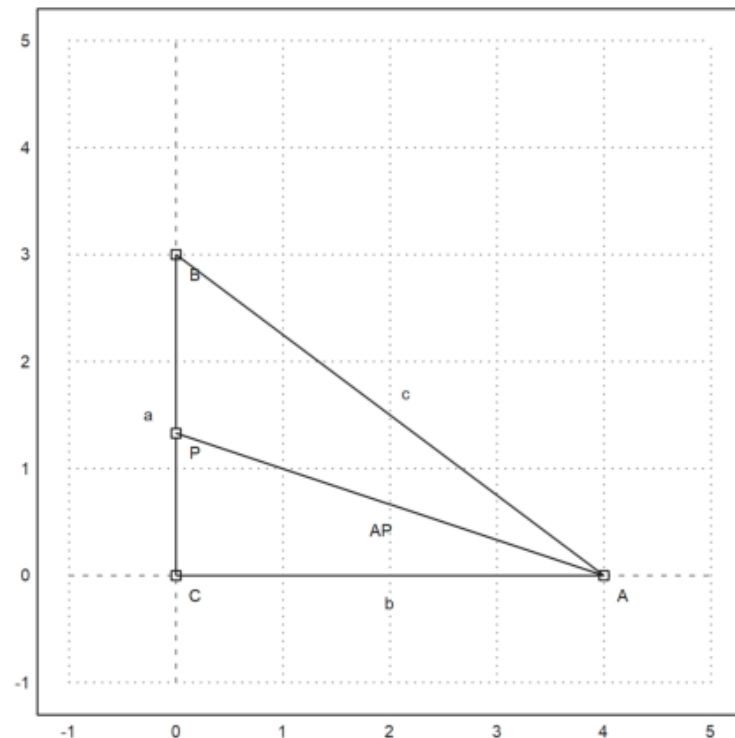
$18^\circ 26' 5.82''$

Titik P adalah perpotongan pembagi sudut dengan sumbu y.

```
>P := [0,tan(wa2)*4]
```

$[0, 1.33333]$

```
>plotPoint(P,"P"); plotSegment(A,P):
```



Mari kita periksa sudut-sudut dalam contoh khusus kita.

```
>computeAngle(C,A,P) , computeAngle(P,A,B)
```

```
0.321750554397
0.321750554397
```

Sekarang kita menghitung panjang pembagi sudut AP.

Kita menggunakan teorema sinus dalam segitiga APC. Teorema ini menyatakan bahwa

$$\frac{BC}{\sin(w_a)} = \frac{AC}{\sin(w_b)} = \frac{AB}{\sin(w_c)}$$

berlaku di segitiga manapun. Kuadratkan, ini diterjemahkan ke dalam hukum yang disebut "spread law"

$$\frac{a}{s_a} = \frac{b}{s_b} = \frac{c}{s_b}$$

di mana a, b, c menunjukkan kuadrans.

Karena spread CPA adalah  $1-sa^2$ , kita mendapatkan dari itu  $bisa/1=b/(1-sa^2)$  dan bisa menghitung bisa (kuadrans dari pembagi sudut).

```
>&factor(ratsimp(b/(1-sa2))); bisa &= %; $bisa
```

$$\frac{2b(b+a)}{\sqrt{b}\sqrt{b+a}+b+a}$$

Mari kita periksa rumus ini untuk nilai-nilai Mesir kita.

```
>sqrt(mxmeval("at(bisa,[a=3^2,b=4^2])"), distance(A,P))
```

4.21637021356

4.21637021356

Kita juga bisa menghitung P menggunakan rumus spread.

```
>py:=factor(ratsimp(sa2*bisa)); $py
```

$$-\frac{b\left(\sqrt{b}\sqrt{b+a}-b-a\right)}{\sqrt{b}\sqrt{b+a}+b+a}$$

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

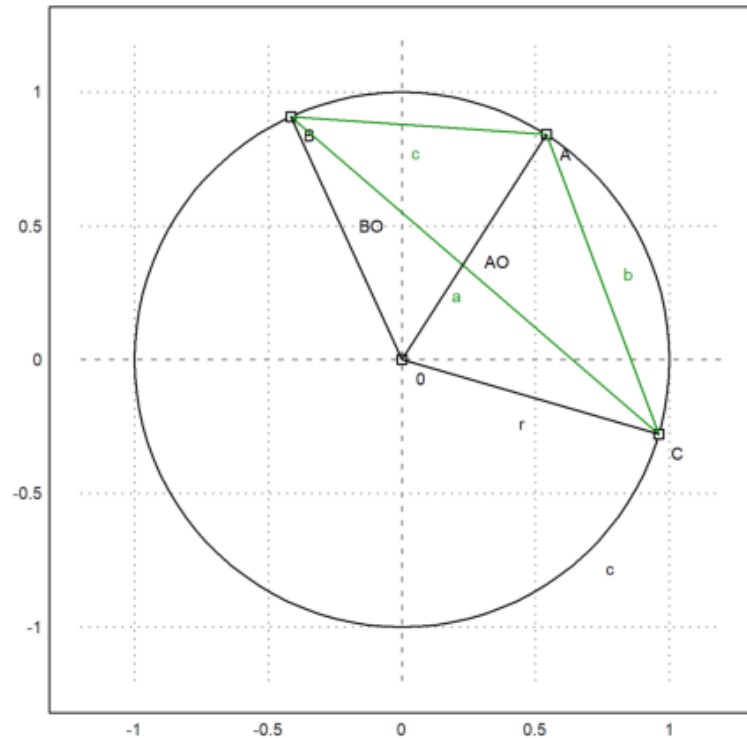
1.33333333333

---

## Sudut Chord

Lihatlah situasi berikut ini.

```
>setPlotRange(1.2); ...
>color(1); plotCircle(circleWithCenter([0,0],1)); ...
>A:=[cos(1),sin(1)]; B:=[cos(2),sin(2)]; C:=[cos(6),sin(6)]; ...
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
>color(3); plotSegment(A,B,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...
>color(1); O:=[0,0]; plotPoint(O,"O"); ...
>plotSegment(A,O); plotSegment(B,O); plotSegment(C,O,"r"); ...
>insimg;
```



Kita bisa menggunakan Maxima untuk menyelesaikan rumus tiga ambang untuk sudut di pusat O untuk  $r$ . Dengan demikian kita mendapatkan rumus untuk radius kuadrat dari perisirkel dalam bentuk kuadrans dari sisi-sisi.

Kali ini, Maxima menghasilkan beberapa nol kompleks, yang kita abaikan.

```
>remvalue(a,b,c,r); // hapus nilai-nilai sebelumnya untuk perhitungan baru
>rabc := rhs(solve(triplespread(spread(b,r,r),spread(a,r,r),spread(c,r,r)),r)[4]); $rabc
```

$$-\frac{abc}{c^2 - 2bc + a(-2c - 2b) + b^2 + a^2}$$

Kita bisa membuatnya menjadi fungsi Euler.

```
>function periradius(a,b,c) := rabc;
```

Mari kita periksa hasilnya untuk titik-titik A, B, C kita.

```
>a:=quadrance(B,C); b:=quadrance(A,C); c:=quadrance(A,B);
```

Jari-jarinya memang 1.

```
>periradius(a,b,c)
```



Faktanya, spread CBA hanya tergantung pada  $b$  dan  $c$ . Ini adalah teorema sudut chord.

```
>$spread(b,a,c)*rabc | ratsimp
```

$$\frac{b}{4}$$

Sebenarnya, spreadnya adalah  $b/(4r)$ , dan kita lihat bahwa sudut chord dari korda  $b$  adalah setengah dari sudut pusat.

```
>$doublespread(b/(4*r))-spread(b,r,r) | ratsimp
```

$$0$$

---

## Contoh 6: Jarak Minimal pada Bidang

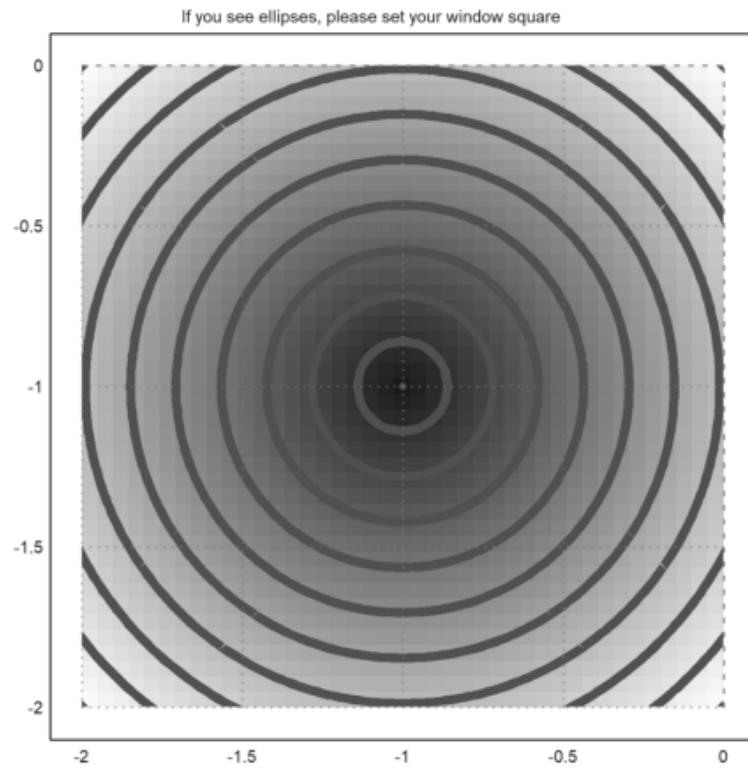
---

### Catatan Awal

---

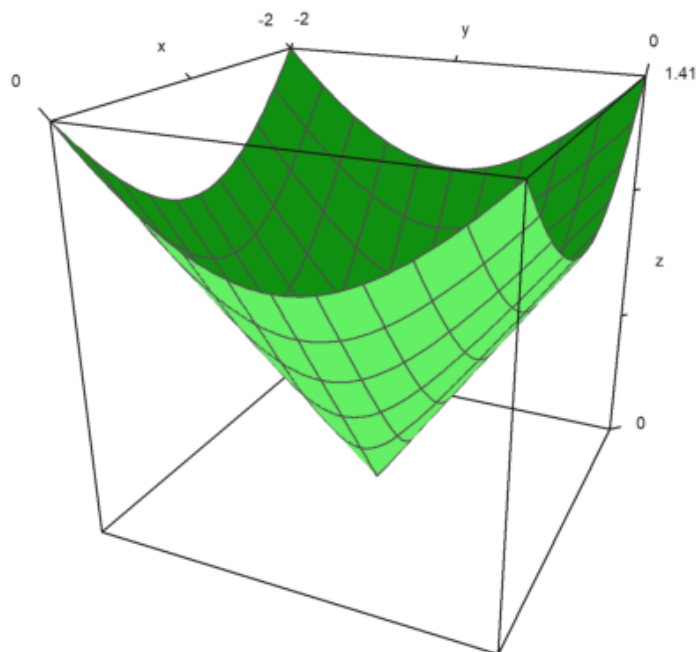
Fungsi yang, pada suatu titik  $M$  dalam bidang, menentukan jarak  $AM$  antara suatu titik tetap  $A$  dan  $M$ , memiliki garis level yang cukup sederhana: lingkaran yang berpusat di  $A$ .

```
>&remvalue();  
>A=[-1,-1];  
>function d1(x,y):=sqrt((x-A[1])^2+(y-A[2])^2)  
>fcontour("d1",xmin=-2,xmax=0,ymin=-2,ymax=0,hue=1, ...  
>title="If you see ellipses, please set your window square");
```



dan grafiknya juga cukup sederhana: bagian atas suatu kerucut:

```
>plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):
```

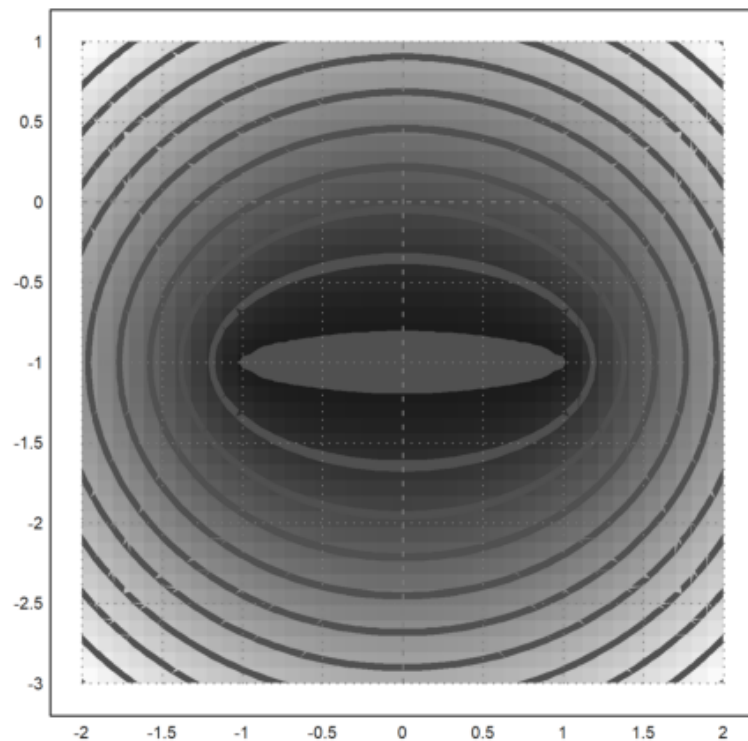


Tentu saja, minimum 0 dicapai di A.

Dua Titik

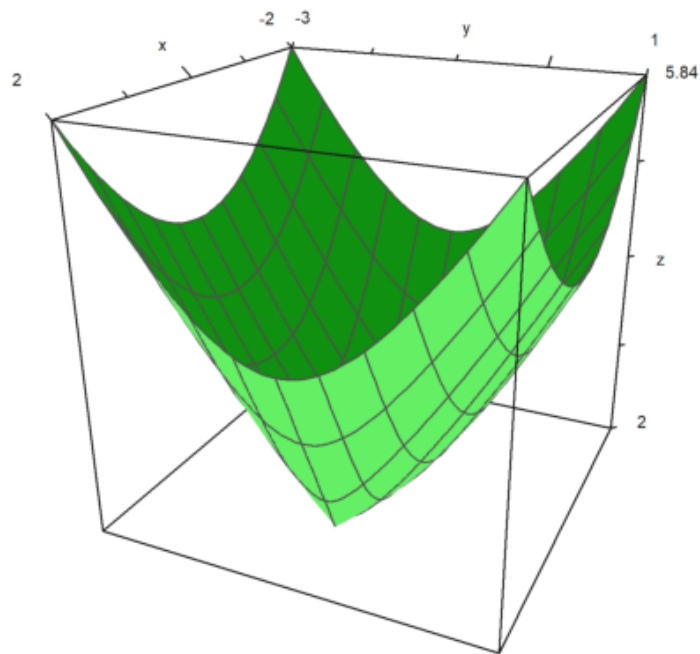
Sekarang kita melihat fungsi  $MA+MB$  di mana A dan B adalah dua titik (tetap). Ini adalah "fakta yang sudah dikenal" bahwa kurva levelnya adalah elips, dengan titik fokus A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

```
>B=[1,-1];  
>function d2(x,y):=d1(x,y)+sqrt((x-B[1])^2+(y-B[2])^2)  
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



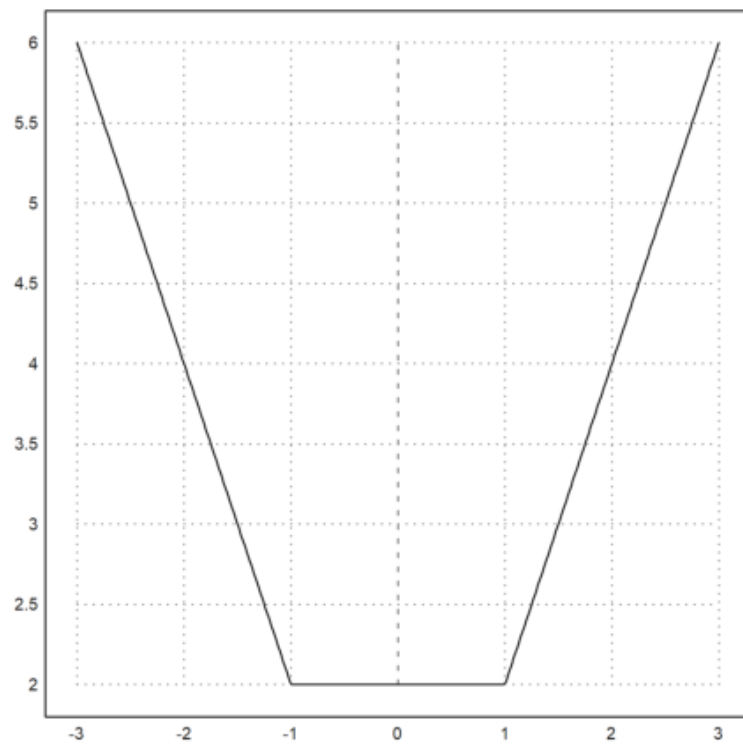
Grafiknya lebih menarik:

```
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```



Pembatasan pada garis (AB) lebih terkenal:

```
>plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
```

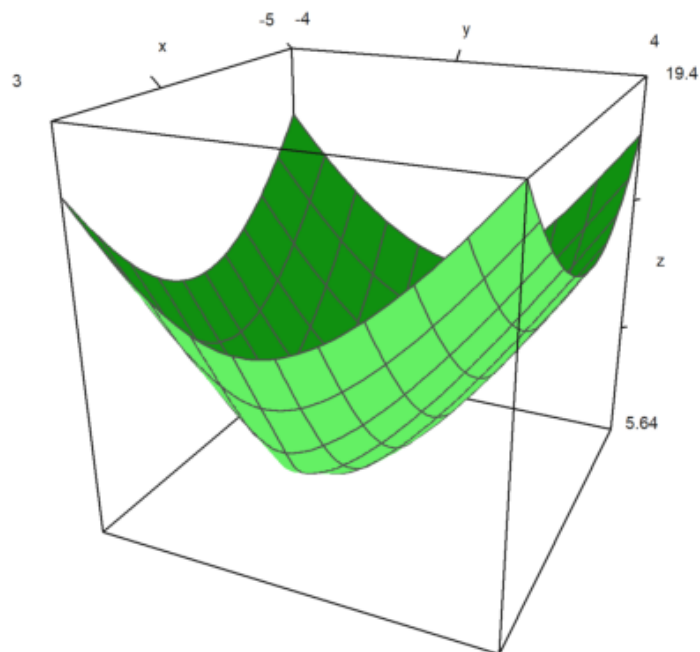


Sekarang hal-hal menjadi kurang sederhana: Sedikit kurang diketahui bahwa  $MA+MB+MC$  mencapai minimumnya di satu titik dalam bidang tetapi menentukannya kurang sederhana:

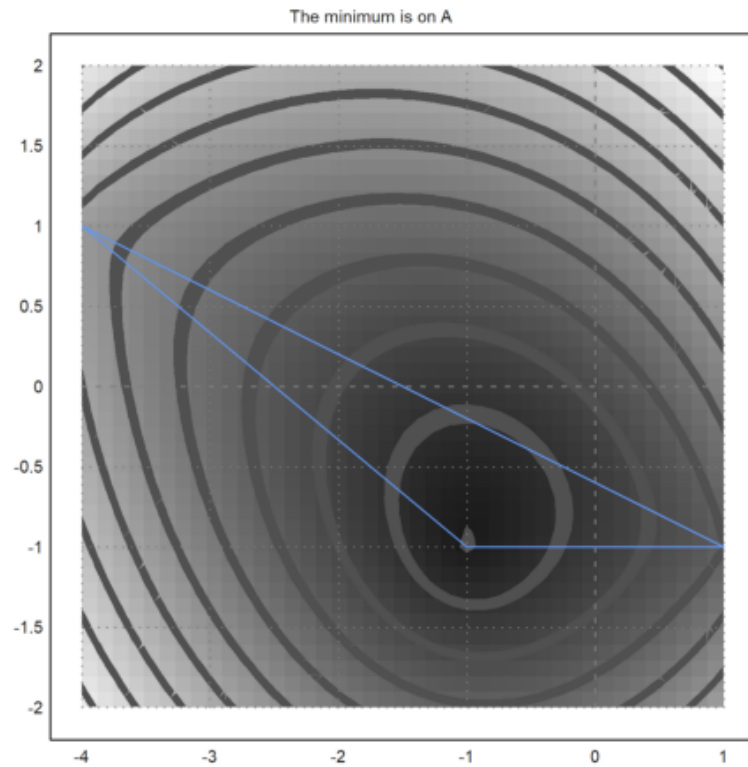
Jika salah satu sudut segitiga ABC lebih dari  $120^\circ$  (katakanlah di A), maka minimum dicapai di titik ini (katakanlah AB+AC).

Contoh:

```
>C=[-4,1];
>function d3(x,y):=d2(x,y)+sqrt((x-C[1])^2+(y-C[2])^2)
>plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);
>insimg;
```

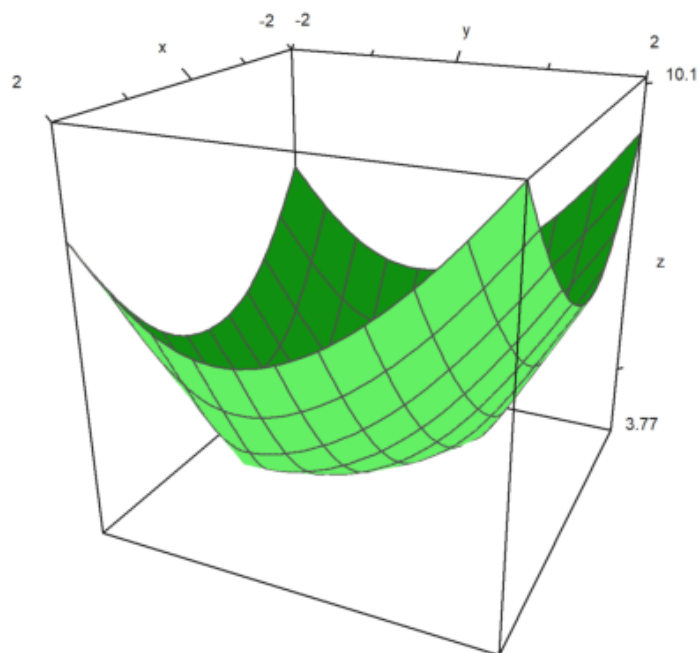


```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

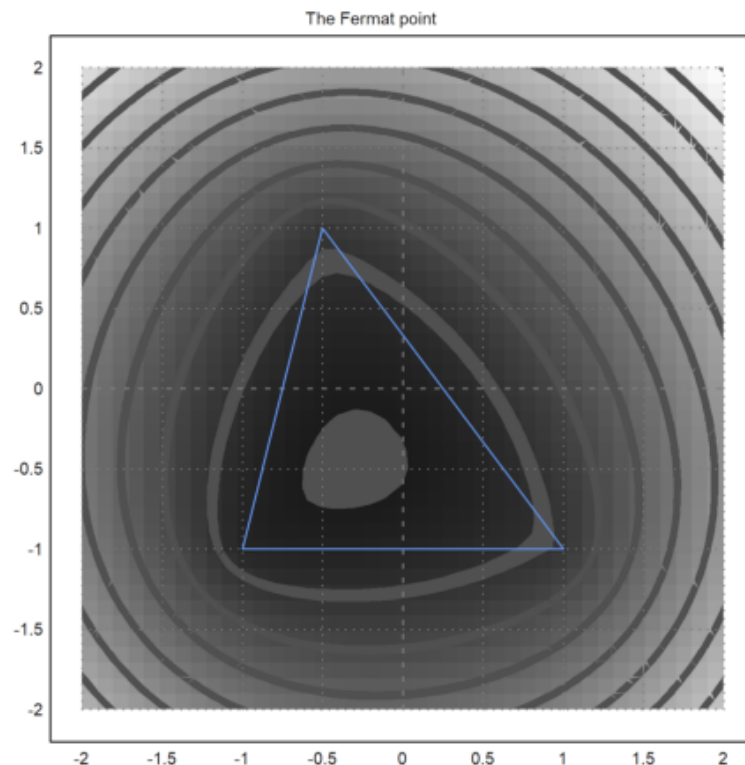


2) Tetapi jika semua sudut segitiga ABC kurang dari  $120^\circ$ , minimum ada di titik F di dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (masing-masing  $120^\circ$ ):

```
>C=[-0.5,1];
>plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):
```



```
>fcontour("d3",xmin=-2,xmax=2,ymin=-2,ymax=2,hue=1,title="The Fermat point");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```



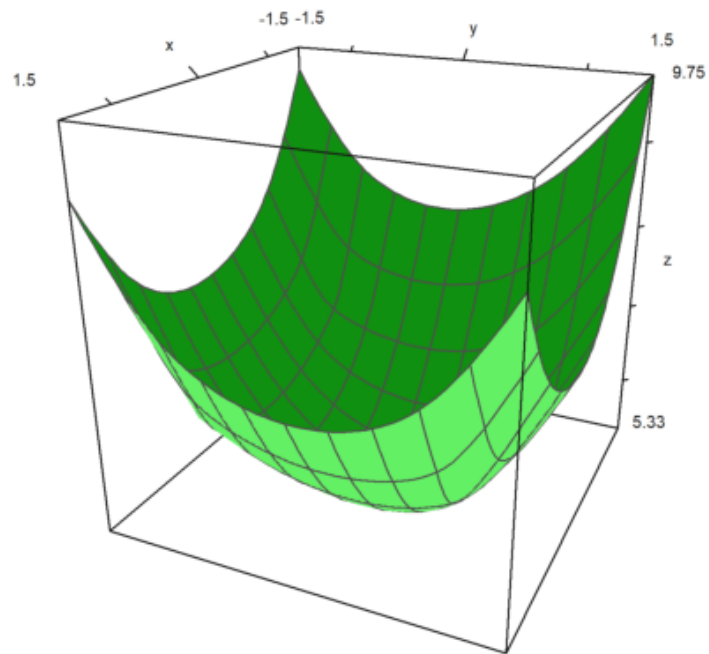
Ini adalah aktivitas menarik untuk merealisasikan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu sebuah perangkat lunak yang ditulis dalam Java yang memiliki instruksi "garis kontur"...

Semua ini telah ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; dia menulis surat kepada dilettante lain seperti imam Marin Mersenne dan Blaise Pascal yang bekerja pada pajak penghasilan. Jadi titik unik F sedemikian rupa sehingga  $FA+FB+FC$  minimal, disebut sebagai titik Fermat dari segitiga. Tetapi tampaknya beberapa tahun sebelumnya, orang Italia Torricelli telah menemukan titik ini sebelum Fermat melakukannya! Bagaimanapun juga, tradisi adalah menandai titik ini F..

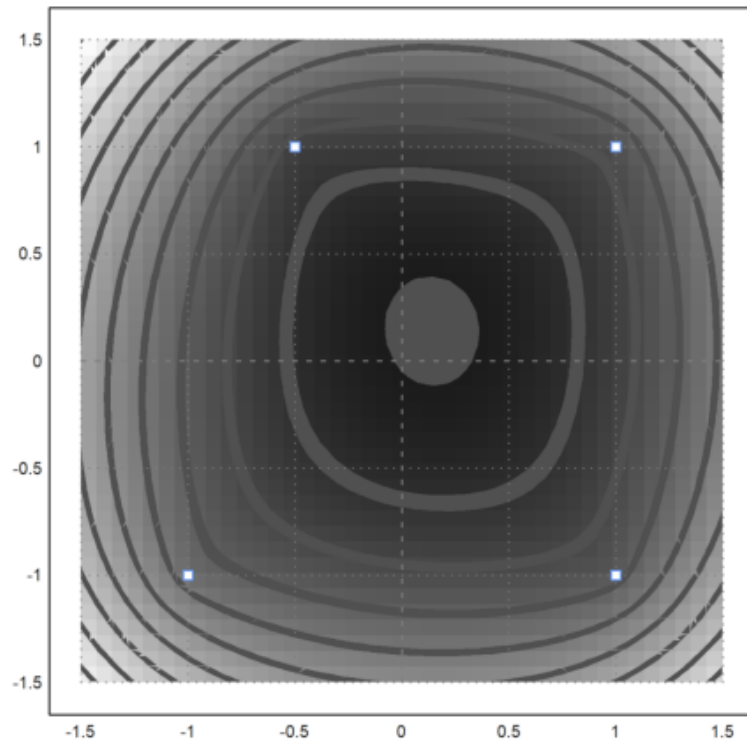
## Empat Titik

Langkah berikutnya adalah menambahkan titik D yang keempat dan mencoba meminimalkan  $MA+MB+MC+MD$ ; katakanlah Anda adalah operator TV kabel dan ingin menemukan di lapangan mana Anda harus menempatkan antenna agar dapat memberi makan empat desa dan menggunakan se sedikit mungkin panjang kabel!

```
>D=[1,1];
>function d4(x,y):=d3(x,y)+sqrt((x-D[1])^2+(y-D[2])^2)
>plot3d("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5):
```



```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
```





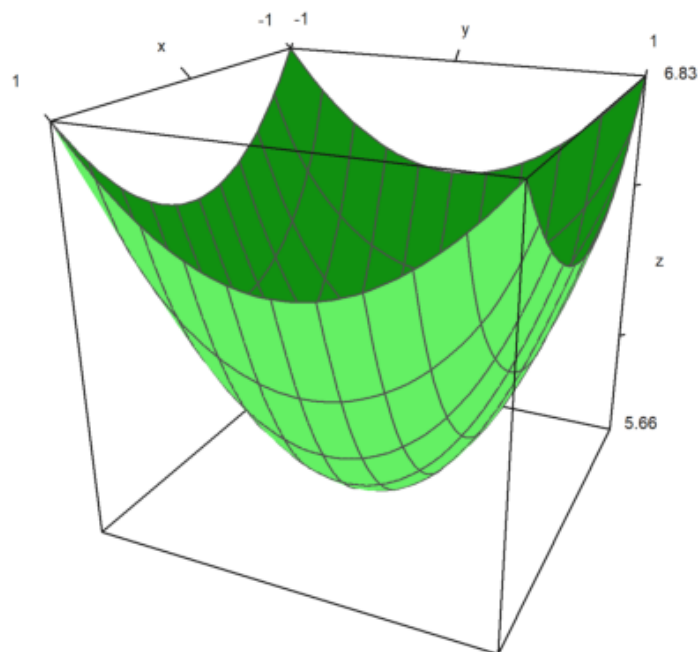
Masih ada minimum dan itu tidak tercapai di salah satu sudut A, B, C, atau D:

```
>function f(x):=d4(x[1],x[2])  
>neldermin("f",[0.2,0.2])
```

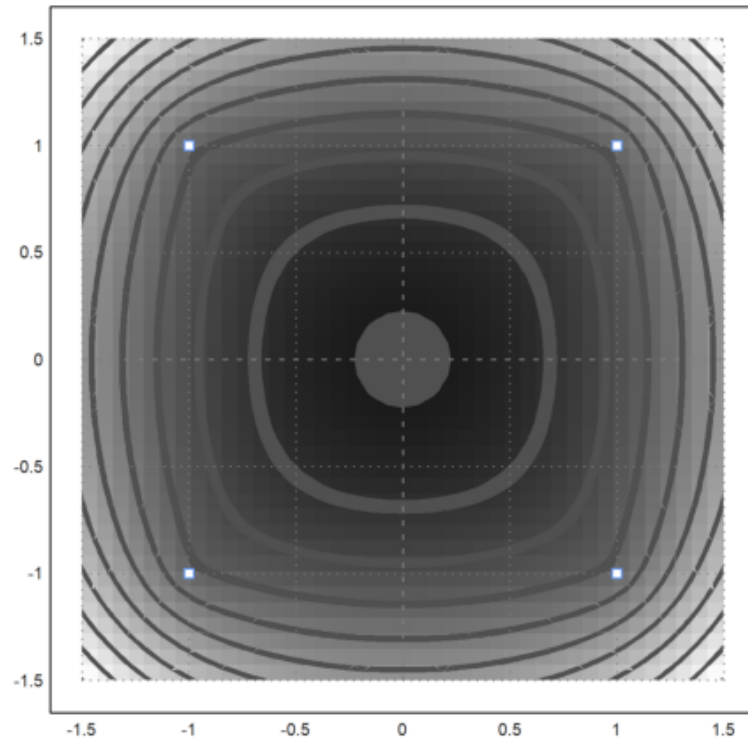
```
[0.142858, 0.142857]
```

Tampaknya bahwa dalam hal ini, koordinat titik optimal bersifat rasional atau mendekati rasional...  
Sekarang ABCD adalah suatu persegi, kita berharap bahwa titik optimal akan menjadi pusat ABCD:

```
>C=[-1,1];  
>plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
```



```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);  
>P=(A_B_C_D)'; plot2d(P[1],P[2],add=1,color=12,points=1);  
>insimg;
```



## Contoh 7: Bola Dandelin dengan Povray

Anda dapat menjalankan demonstrasi ini jika Anda telah menginstal Povray, dan pvengine.exe berada dalam jalur program.

Pertama, kita menghitung jari-jari bola.

Jika Anda melihat gambar di bawah ini, Anda akan melihat bahwa kita memerlukan dua lingkaran yang menyentuh dua garis yang membentuk kerucut, dan satu garis yang membentuk bidang pemotongan kerucut. Kami menggunakan file geometry.e dari Euler untuk ini.

```
>load geometry;
```

Pertama bentuk dua garis membentuk kerucut.

```
>g1 &= lineThrough([0,0],[1,a])
```

$$[-a, 1, 0]$$

```
>g2 &= lineThrough([0,0],[-1,a])
```

$$[-a, -1, 0]$$

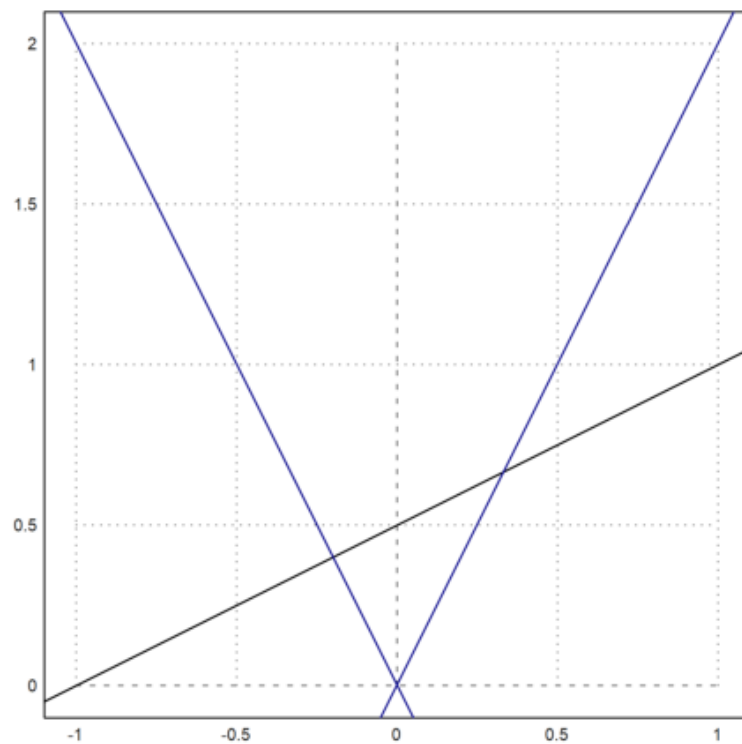
Kemudian beri garis ke tiga

```
>g &= lineThrough([-1,0],[1,1])
```

$[-1, 2, 1]$

Kami merencanakan semua yang sudah terjadi.

```
>setPlotRange(-1,1,0,2);  
>color(black); plotLine(g(),"")  
>a:=2; color(blue); plotLine(g1(),""), plotLine(g2(),""):
```



Sekarang kita ambil titik umum pada sumbu y.

```
>P &= [0,u]
```

$[0, u]$

Menghitung jarak ke g1

```
>d1 &= distance(P,projectToLine(P,g1)); $d1
```

$$\sqrt{\left(\frac{a^2 u}{a^2 + 1} - u\right)^2 + \frac{a^2 u^2}{(a^2 + 1)^2}}$$

Menghitung jarak ke g.

```
>d &= distance(P,projectToLine(P,g)); $d
```

$$\sqrt{\left(\frac{u+2}{5} - u\right)^2 + \frac{(2u-1)^2}{25}}$$

Menentukan pusat kedua lingkaran yang jaraknya sama.

```
>sol &= solve(d1^2=d^2,u); $sol
```

$$\left[ u = \frac{-\sqrt{5}\sqrt{a^2+1}+2a^2+2}{4a^2-1}, u = \frac{\sqrt{5}\sqrt{a^2+1}+2a^2+2}{4a^2-1} \right]$$

Ada dua solusi.

Kita mengevaluasi solusi simbolis, dan menemukan kedua pusat, serta kedua jarak.

```
>u := sol()
```

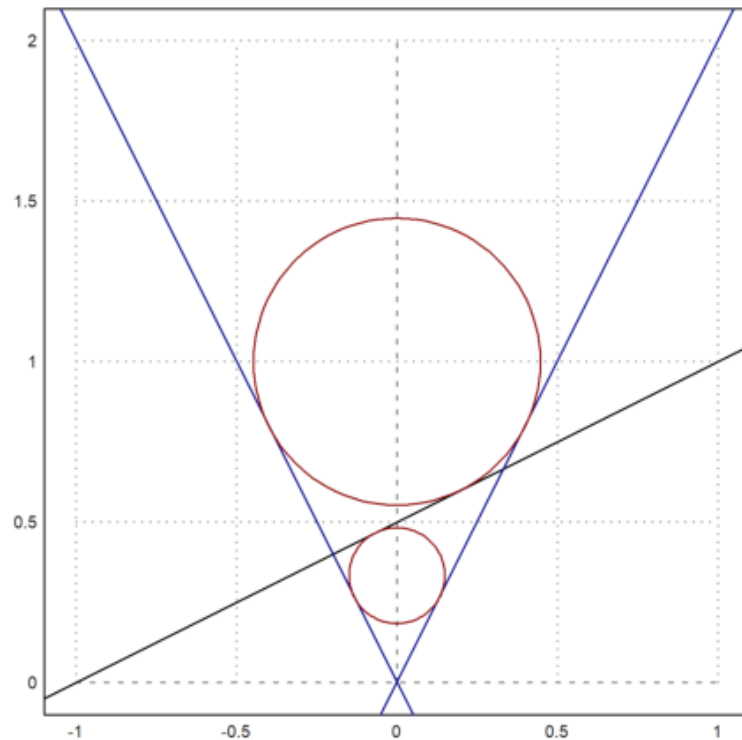
```
[0.333333, 1]
```

```
>dd := d()
```

```
[0.149071, 0.447214]
```

Gambarkan lingkaran ke dalam gambar.

```
>color(red);  
>plotCircle(circleWithCenter([0,u[1]],dd[1]), "");  
>plotCircle(circleWithCenter([0,u[2]],dd[2]), "");  
>insimg;
```



## Plot dengan Povray

Selanjutnya, kita membuat plot dengan Povray. Perhatikan bahwa kita dapat mengubah perintah apa pun dalam urutan perintah Povray berikut, dan menjalankan ulang semua perintah dengan Shift-Return.

Pertama, kita muat fungsi-fungsi povray.

```
>load povray;
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"
```

```
C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe
```

Kemudian, kita atur perintah dengan tepat.

```
>povstart (zoom=11,center=[0,0,0.5],height=10°,angle=140°);
```

Selanjutnya, kita tulis dua bola ke file Povray.

```
>writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
>writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
```

Dan kerucutnya, berwarna transparan.

```
>writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
```

Kita hasilkan bidang terbatas pada kerucut.

```
>gp=g();
>pc=povcone([0,0,0],0,[0,0,a],1,"");
>vp=[gp[1],0,gp[2]]; dp=gp[3];
>writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```

Sekarang kita menghasilkan dua titik di lingkaran, di mana bola menyentuh kerucut.

```
>function turnz(v) := return [-v[2],v[1],v[3]]
>P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
>writeln(povpoint(P1,povlook(yellow)));
>P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
>writeln(povpoint(P2,povlook(yellow)));
```

Kemudian kita menghasilkan dua titik di mana bola menyentuh bidang. Ini adalah fokus dari elips.

```
>P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
>writeln(povpoint(P3,povlook(yellow)));
>P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
>writeln(povpoint(P4,povlook(yellow)));
```

Selanjutnya kita hitung irisan dari P1 dan P2 dengan bidang.

```
>t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1);
>writeln(povpoint(P5,povlook(yellow)));
```

Kita hubungkan titik-titik dengan segmen garis.

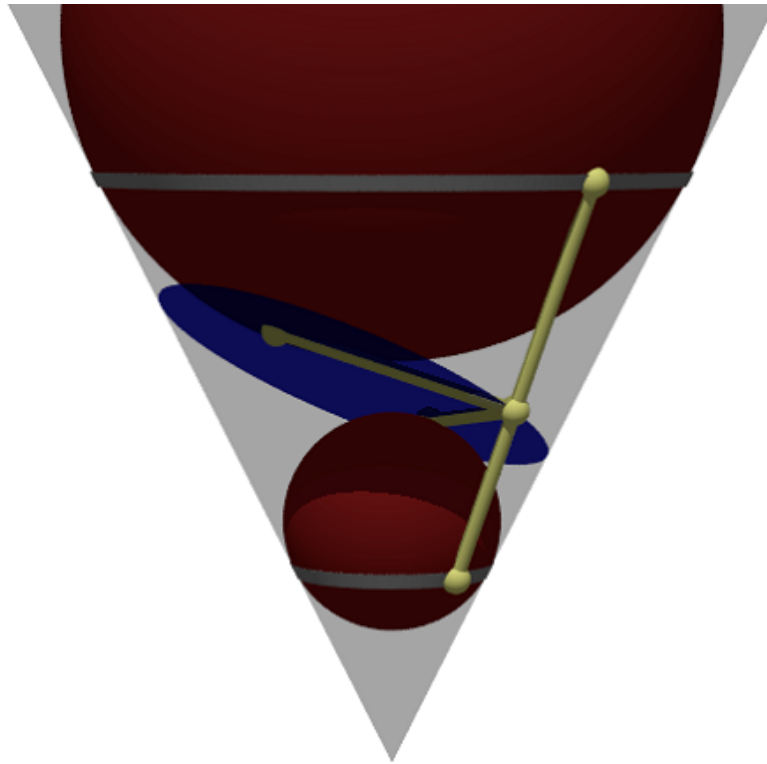
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan pita abu-abu, di mana bola menyentuh kerucut.

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);
>pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
>writeln(povintersection([pcw,pc1],povlook(gray)));
>pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

Memulai program Povray.

```
>povend();
```

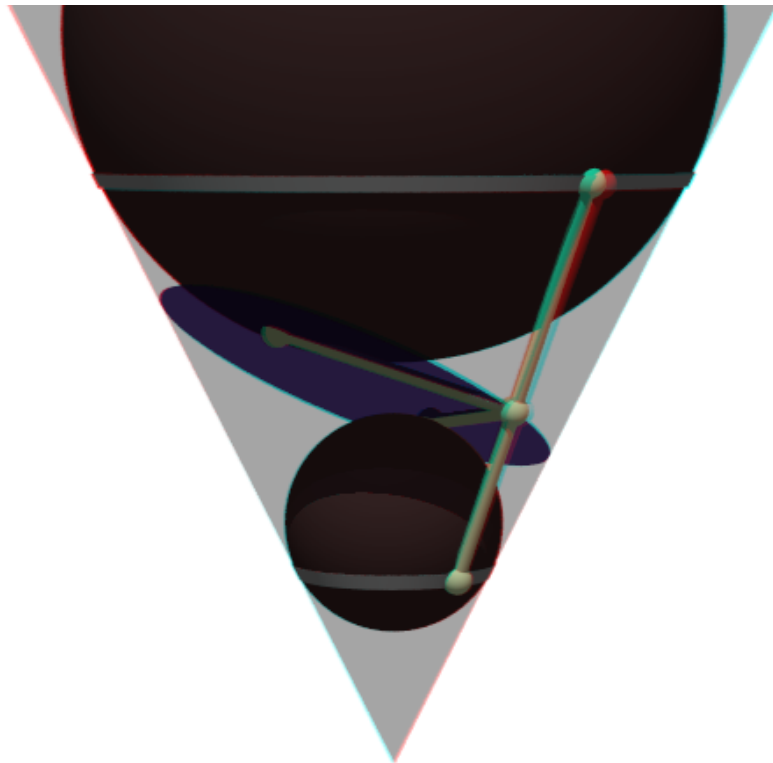


Untuk mendapatkan Anaglyph dari ini, kita perlu memasukkan semuanya ke dalam fungsi adegan. Fungsi ini akan digunakan dua kali nanti.

```
>function scene () ...
```

```
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g();
pc=povcone([0,0,0],0,[0,0,a],1,"");
vp=[gp[1],0,gp[2]]; dp=gp[3];
writeln(povplane(vp,dp,povlook(blue,0.5),pc));
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
writeln(povpoint(P1,povlook(yellow)));
P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
writeln(povpoint(P2,povlook(yellow)));
P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
writeln(povpoint(P3,povlook(yellow)));
P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
writeln(povpoint(P4,povlook(yellow)));
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1);
writeln(povpoint(P5,povlook(yellow)));
writeln(povsegment(P1,P2,povlook(yellow)));
```

```
>povanaglyph("scene",zoom=11,center=[0,0,0.5],height=10°,angle=140°);
```



Dalam notebook ini, kita ingin melakukan beberapa perhitungan bola. Fungsi-fungsi ini terdapat dalam file "spherical.e" di folder contoh. Kami perlu memuat file tersebut terlebih dahulu.

Untuk memasukkan posisi geografis, kami menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut adalah koordinat untuk Kampus FMIPA UNY.

$$[-0.13569, \quad 1.92657]$$



Anda dapat mencetak posisi ini dengan menggunakan sposprint (cetak posisi bola).

```
>sposprint(FMIPA) // posisi garis lintang dan garis bujur FMIPA UNY
```

```
S 7°46.467' E 110°23.050'
```

Mari tambahkan dua kota lagi, Solo dan Semarang.

```
>Solo=[rad(-7,-34.333),rad(110,49.683)]; Semarang=[rad(-6,-59.05),rad(110,24.533)];  
>sposprint(Solo), sposprint(Semarang),
```

```
S 7°34.333' E 110°49.683'  
S 6°59.050' E 110°24.533'
```

Pertama-tama kita menghitung vektor dari satu kota ke kota lainnya pada bola ideal. Vektor ini adalah [heading, jarak] dalam radian. Untuk menghitung jarak di Bumi, kita mengalikannya dengan jari-jari Bumi pada lintang  $7^\circ$ .

```
>br=svector(FMIPA,Solo); degprint(br[1]), br[2]*rearth(7°)->km // perkiraan jarak FMIPA-Solo
```

```
65°20'26.60''  
53.8945384608
```

Ini adalah perkiraan yang baik. Rutinitas-rutinitas berikut menggunakan perkiraan yang lebih baik lagi. Pada jarak yang begitu pendek, hasilnya hampir sama.

```
>esdist(FMIPA,Semarang)->"km"
```

```
88.0114026318km
```

Ada fungsi untuk heading, yang memperhitungkan bentuk elips Bumi. Sekali lagi, kami mencetaknya dengan cara yang lebih canggih.

```
>sdegprint(esdir(FMIPA,Solo))
```

```
65.34°
```

Sudut dari segitiga melebihi  $180^\circ$  pada bola.

```
>asum=sangle(Solo,FMIPA,Semarang)+sangle(FMIPA,Solo,Semarang)+sangle(FMIPA,Semarang,Solo);
```

```
180°0'10.77''
```

Ini dapat digunakan untuk menghitung luas segitiga. Catatan: Untuk segitiga yang kecil, ini tidak akurat karena kesalahan pengurangan dalam asum-pi.

```
>(asum-pi)*rearth(48°)^2->" km^2"
```

```
2116.02948749 km^2
```

Ada fungsi untuk ini, yang menggunakan lintang rata-rata segitiga untuk menghitung jari-jari Bumi, dan mengatasi kesalahan pembulatan untuk segitiga yang sangat kecil.

```
>esarea(Solo,FMIPA,Semarang)->" km^2", //perkiraan yang sama dengan fungsi esarea()
```

```
2123.64310526 km^2
```

Kita juga bisa menambahkan vektor ke posisi. Sebuah vektor berisi heading dan jarak, keduanya dalam radian. Untuk mendapatkan vektor, kita menggunakan `svector`. Untuk menambahkan vektor ke posisi, kita menggunakan `saddvector`.

```
>v=svector(FMIPA,Solo); sposprint(saddvector(FMIPA,v)), sposprint(Solo),
```

```
S 7°34.333' E 110°49.683'  
S 7°34.333' E 110°49.683'
```

Fungsi-fungsi ini mengasumsikan bola ideal. Sama dengan di Bumi.

```
>sposprint(esadd(FMIPA,esdir(FMIPA,Solo),esdist(FMIPA,Solo))), sposprint(Solo),
```

```
S 7°34.333' E 110°49.683'  
S 7°34.333' E 110°49.683'
```

Mari kita beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

```
>Tugu=[-7.7833°,110.3661°]; Monas=[-6.175°,106.811944°];  
>sposprint(Tugu), sposprint(Monas)
```

```
S 7°46.998' E 110°21.966'  
S 6°10.500' E 106°48.717'
```

Menurut Google Earth, jaraknya adalah 429.66 km. Kami mendapatkan perkiraan yang baik.

```
>esdist(Tugu,Monas)->"km"
```

```
431.565659488km
```

Heading-nya sama dengan yang dihitung di Google Earth.

```
>degprint(esdir(Tugu,Monas))
```

```
294°17'2.85''
```

Namun, kita tidak lagi mendapatkan posisi target yang tepat, jika kita menambahkan heading dan jarak ke posisi asli. Ini terjadi karena kita tidak menghitung fungsi inversnya dengan tepat, tetapi mengambil perkiraan jari-jari Bumi sepanjang jalur.

```
>sposprint (esadd(Tugu,esdir (Tugu,Monas) , esdist (Tugu,Monas) ) )
```

S 6°10.500' E 106°48.717'

Kesalahan ini tidak besar, bagaimanapun.

```
>sposprint (Monas) ,
```

S 6°10.500' E 106°48.717'

Tentu saja, kita tidak dapat berlayar dengan heading yang sama dari satu tujuan ke tujuan lain, jika kita ingin mengambil jalur terpendek. Bayangkan, Anda terbang ke timur laut dari titik mana pun di Bumi. Kemudian Anda akan berputar-putar ke kutub utara. Garis besar tidak mengikuti heading konstan!

Perhitungan berikut menunjukkan bahwa kita jauh dari tujuan yang benar, jika kita menggunakan heading yang sama selama perjalanan.

```
>dist=esdist (Tugu,Monas) ; hd=esdir (Tugu,Monas) ;
```

Sekarang kita tambahkan 10 kali satu persepuluh jarak, menggunakan heading ke Monas yang kita dapatkan di Tugu.

```
>p=Tugu; loop 1 to 10; p=esadd(p,hd,dist/10); end;
```

Hasilnya jauh dari benar.

```
>sposprint (p) , skmprint (esdist (p,Monas) )
```

S 6°11.250' E 106°48.372'  
1.529km

Sebagai contoh lain, mari kita ambil dua titik di Bumi pada lintang yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

alur terpendek dari P1 ke P2 bukanlah lingkaran lintang 30°, tetapi jalur yang lebih pendek dimulai 10° lebih ke utara di P1.

```
>sdegprint (esdir (P1,P2) )
```

79.69°

Tetapi, jika kita mengikuti bacaan kompas ini, kita akan berputar-putar ke kutub utara! Jadi kita harus menyesuaikan heading kita sepanjang jalan. Untuk tujuan kasar, kita menyesuaikannya setiap 1/10 dari jarak total.

```
>p=P1; dist=esdist(P1,P2); ...
> loop 1 to 10; dir=esdir(p,P2); sdegprint(dir), p=esadd(p,dir,dist/10); end;
```

```
79.69°
81.67°
83.71°
85.78°
87.89°
90.00°
92.12°
94.22°
96.29°
98.33°
```

Jaraknya tidak benar, karena kita akan menambahkan sedikit kesalahan jika kita mengikuti heading yang sama terlalu lama.

```
>skmprint(esdist(p,P2))
```

```
0.203km
```

We get a good approximation, if we adjust our heading after each 1/100 of the total distance from Tugu to Monas.

```
>p=Tugu; dist=esdist(Tugu,Monas); ...
> loop 1 to 100; p=esadd(p,esdir(p,Monas),dist/100); end;
>skmprint(esdist(p,Monas))
```

```
0.000km
```

Untuk tujuan navigasi, kita bisa mendapatkan rangkaian posisi GPS sepanjang garis besar ke Monas dengan fungsi `navigate`.

```
>load spherical; v=navigate(Tugu,Monas,10); ...
> loop 1 to rows(v); sposprint(v[#]), end;
```

```
S 7°46.998' E 110°21.966'
S 7°37.422' E 110°0.573'
S 7°27.829' E 109°39.196'
S 7°18.219' E 109°17.834'
S 7°8.592' E 108°56.488'
S 6°58.948' E 108°35.157'
S 6°49.289' E 108°13.841'
S 6°39.614' E 107°52.539'
S 6°29.924' E 107°31.251'
S 6°20.219' E 107°9.977'
S 6°10.500' E 106°48.717'
```

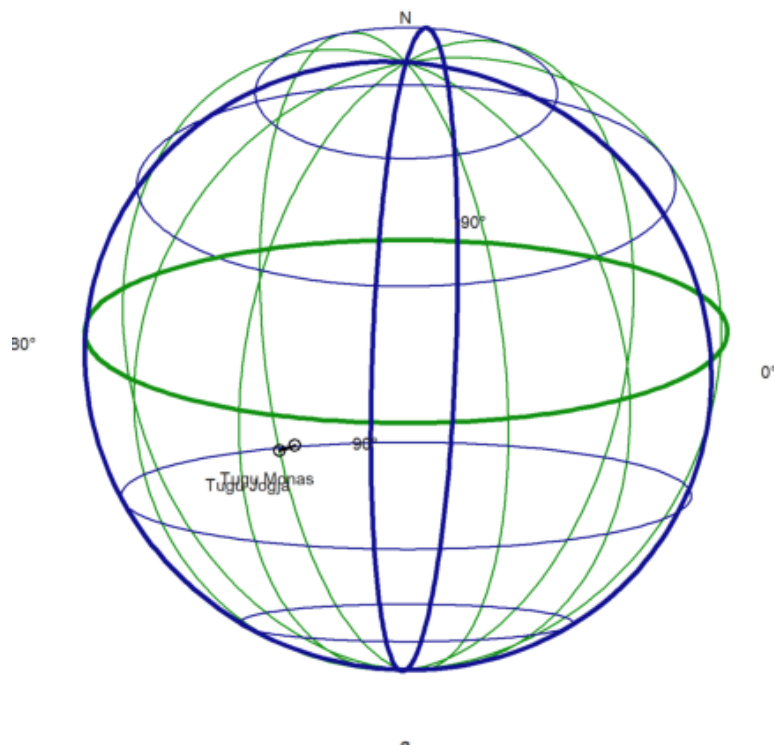
Kita menulis sebuah fungsi yang memplotkan Bumi, dua posisi, dan posisi di antaranya.

```
>function testplot ...
```

```
useglobal;  
plotearth;  
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");  
plotposline(v);  
endfunction
```

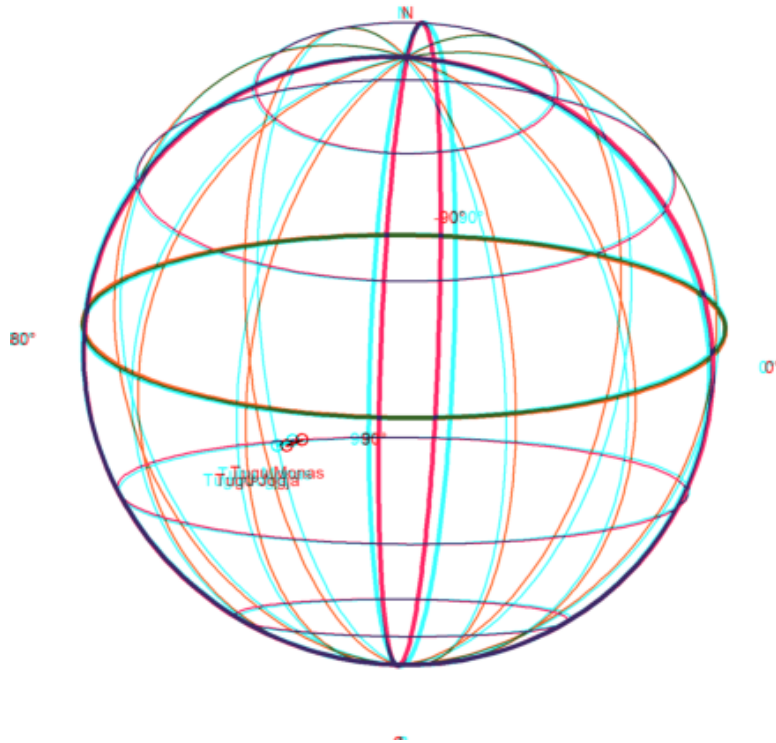
Sekarang plot semuanya.

```
>plot3d("testplot",angle=25, height=6,>own,>user, zoom=4):
```



Atau gunakan plot3d untuk mendapatkan tampilan anaglyph. Ini terlihat sangat bagus dengan kacamata merah/cyan.

```
>plot3d("testplot",angle=25,height=6,distance=5,own=1,anaglyph=1, zoom=4):
```



## Latihan

1. Gambarlah segi-n beraturan jika diketahui titik pusat  $O$ ,  $n$ , dan jarak titik pusat ke titik-titik sudut segi-n tersebut (jari-jari lingkaran luar segi-n),  $r$ .

Petunjuk:

- Besar sudut pusat yang menghadap masing-masing sisi segi-n adalah  $(360/n)$ .
- Titik-titik sudut segi-n merupakan perpotongan lingkaran luar segi-n dan garis-garis yang melalui pusat dan saling membentuk sudut sebesar kelipatan  $(360/n)$ .
- Untuk  $n$  ganjil, pilih salah satu titik sudut adalah di atas.
- Untuk  $n$  genap, pilih 2 titik di kanan dan kiri lurus dengan titik pusat.
- Anda dapat menggambar segi-3, 4, 5, 6, 7, dst beraturan.

2. Gambarlah suatu parabola yang melalui 3 titik yang diketahui.

Petunjuk:

- Misalkan persamaan parabolanya  $y = ax^2 + bx + c$ .
- Substitusikan koordinat titik-titik yang diketahui ke persamaan tersebut.
- Selesaikan SPL yang terbentuk untuk mendapatkan nilai-nilai  $a$ ,  $b$ ,  $c$ .

3. Gambarlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya  $A$ ,  $B$ ,  $C$ ,  $D$ .

- Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung (sisinya-sisintya merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).

- Suatu segi-4 merupakan segi-4 garis singgung apabila keempat garis bagi sudutnya bertemu di satu titik.

- Jika segi-4 tersebut merupakan segi-4 garis singgung, gambar lingkaran dalamnya.
- Tunjukkan bahwa syarat suatu segi-4 merupakan segi-4 garis singgung apabila hasil kal

panjang sisi-sisi yang berhadapan sama.

4. Gambarlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).
5. Gambarlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).