

# 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  
setPlotrange(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

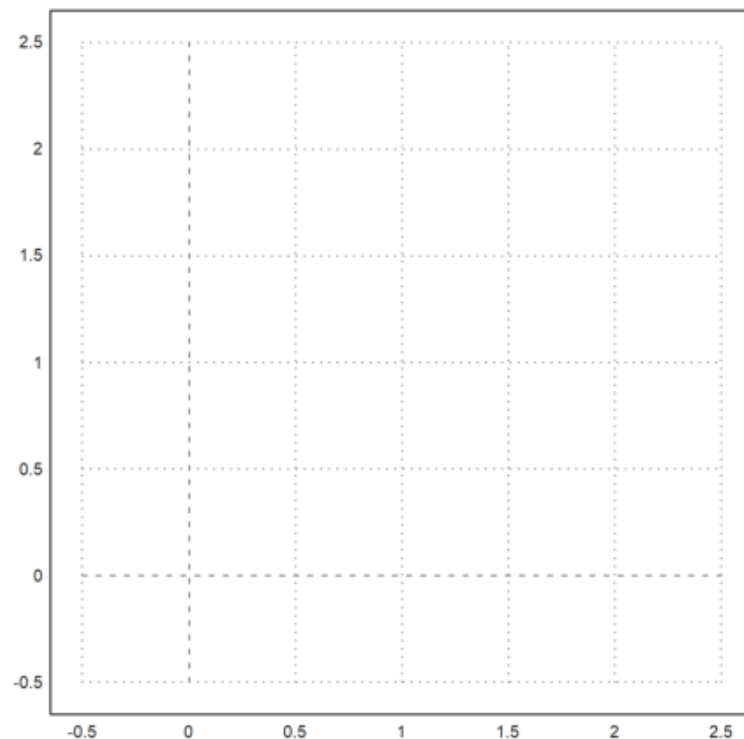
$sa = \sin(\phi)^2 \text{ spread } a$ .

## 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
```

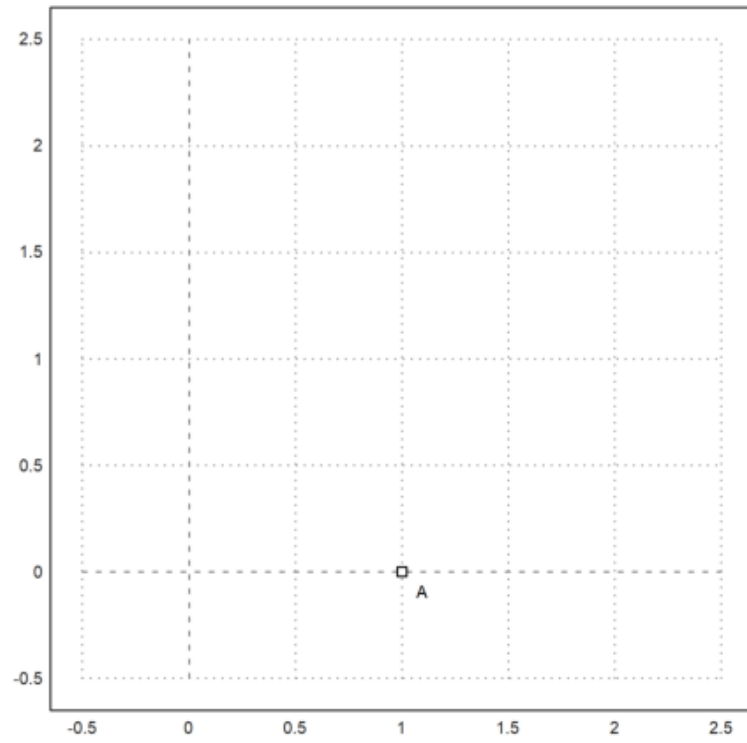


## Menggambar Titik dan Ruas Garis

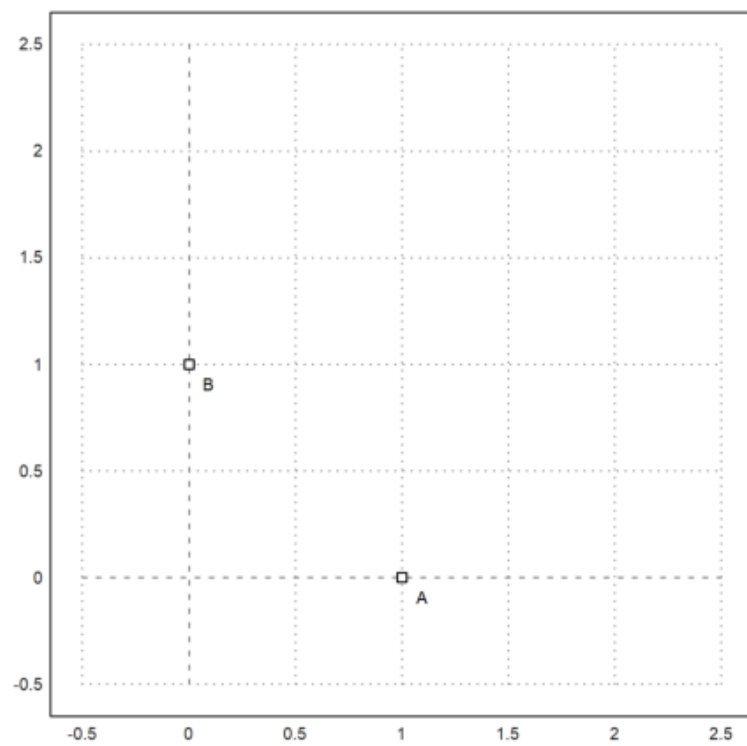
---

Sekarang tetapkan tiga titik dan plot mereka. Kita ambil 3 titik A,B,C, misalkan titik A pada koordinat (1,0)

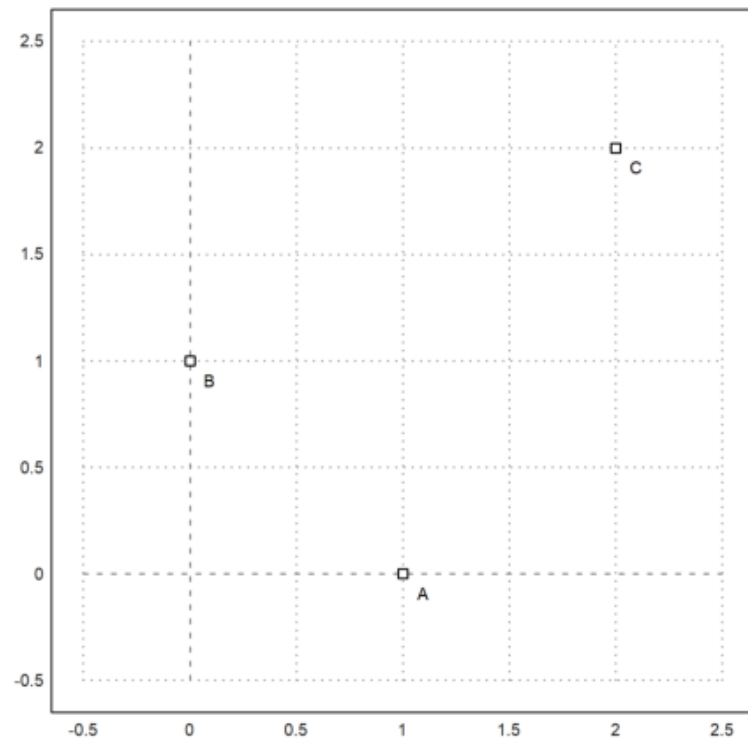
```
>A=[1,0]; plotPoint(A,"A"): // definisi dan gambar tiga titik
```



```
>B=[0,1]; plotPoint(B,"B"):
```

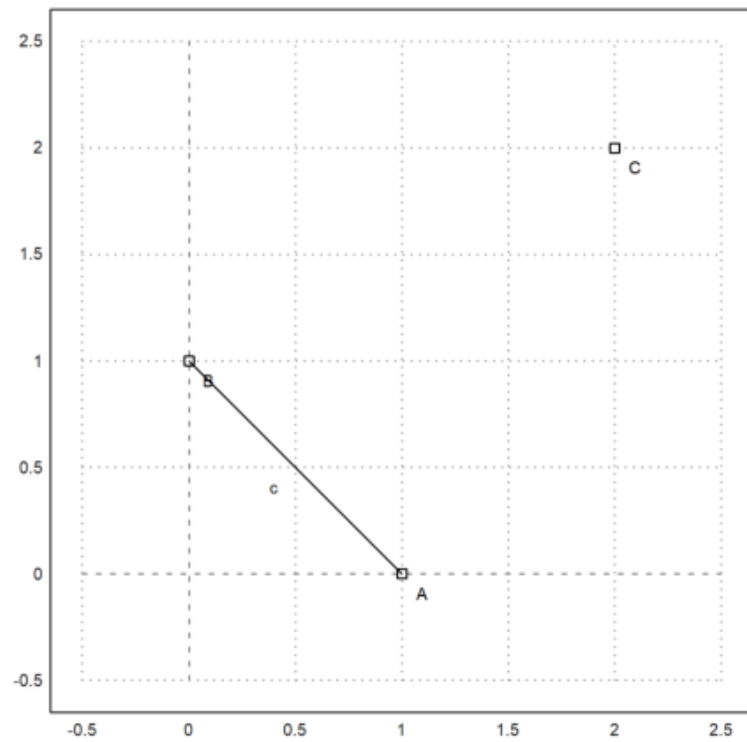


```
>C=[2,2]; plotPoint(C,"C"):
```

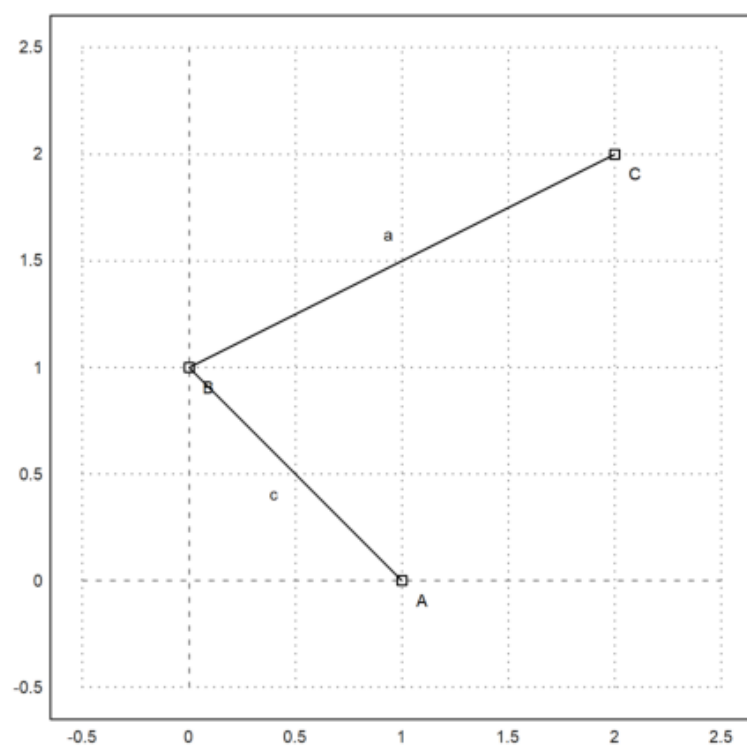


Kemudian kita akan menghubungkan titik A dan B menjadi sebuah garis yang diberi nama "c"

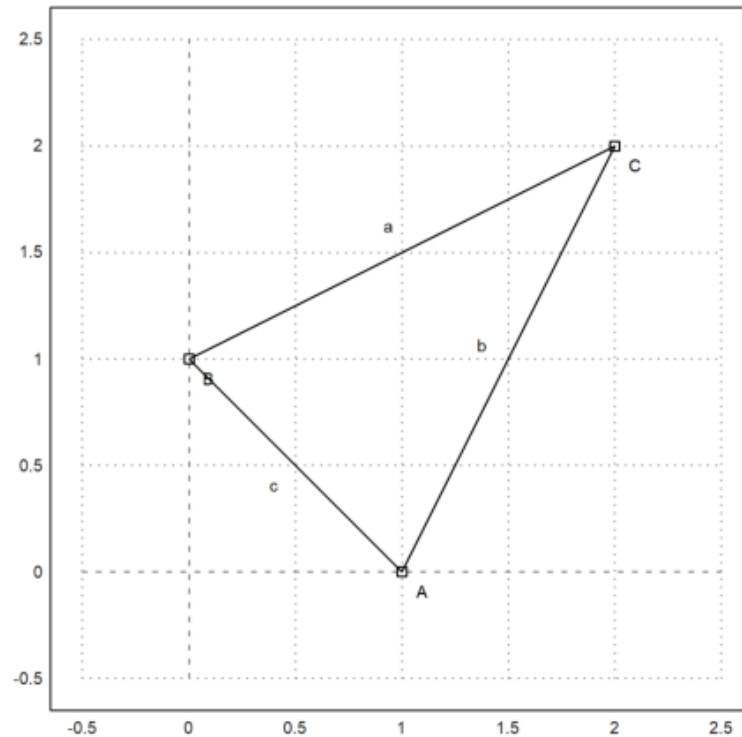
```
>plotSegment(A,B,"c"): // c=AB
```



```
>plotSegment(B,C,"a"): // a=BC
```



```
>plotSegment(A,C,"b"): // b=AC
```



Setelah kita menghubungkan setiap titiknya diperoleh suatu bidang berupa segitiga "ABC" dengan ruas garis a,b,c.

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

### Menggambar Garis yang Melalui 2 Titik

---

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

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

Setelah diperoleh garis melalui B dan C, kita cari garis tegak lurus yang melalui A pada BC

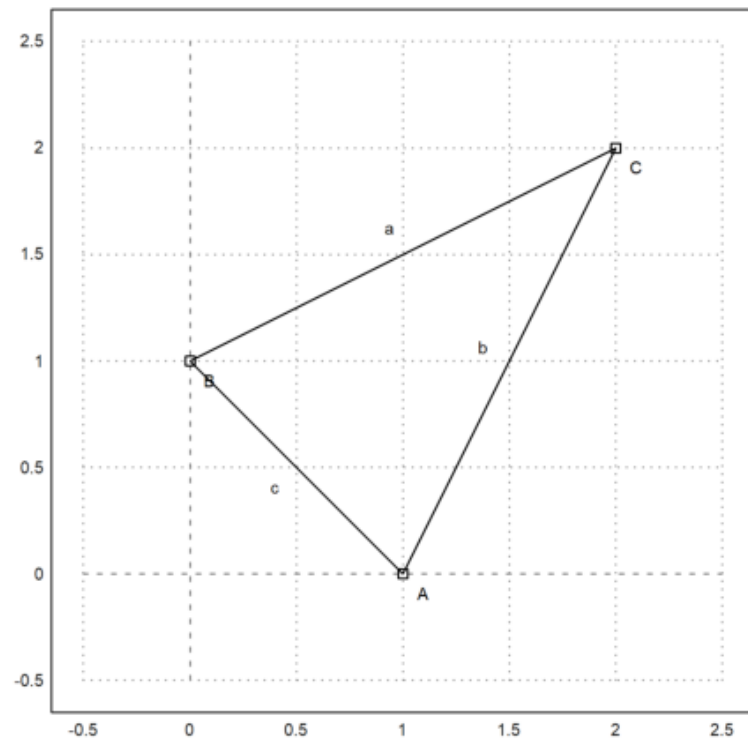
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan persimpangannya dengan BC.

### Menggambar Titik Potong

---

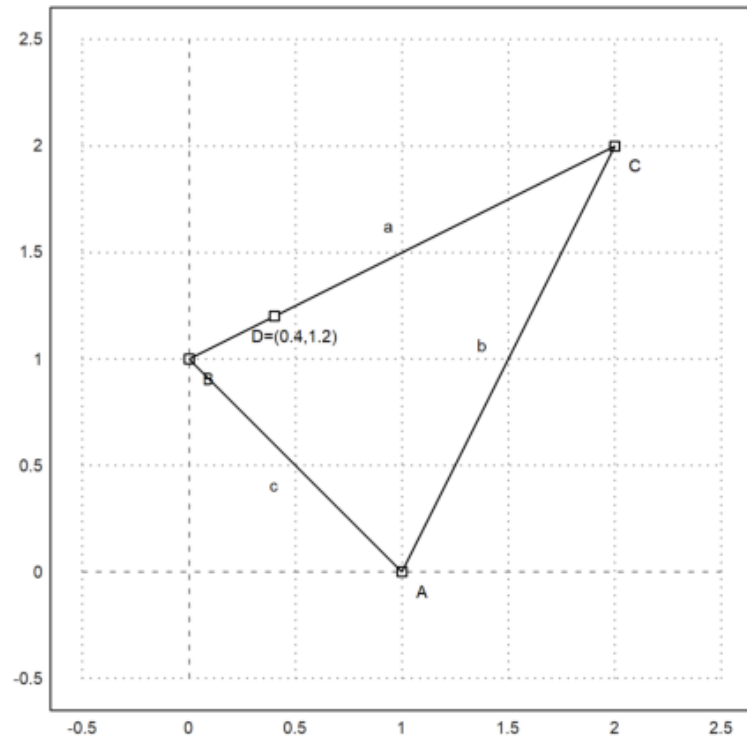
```
>D=lineIntersection(h,lineThrough(B,C)): // D adalah titik potong h dan BC
```



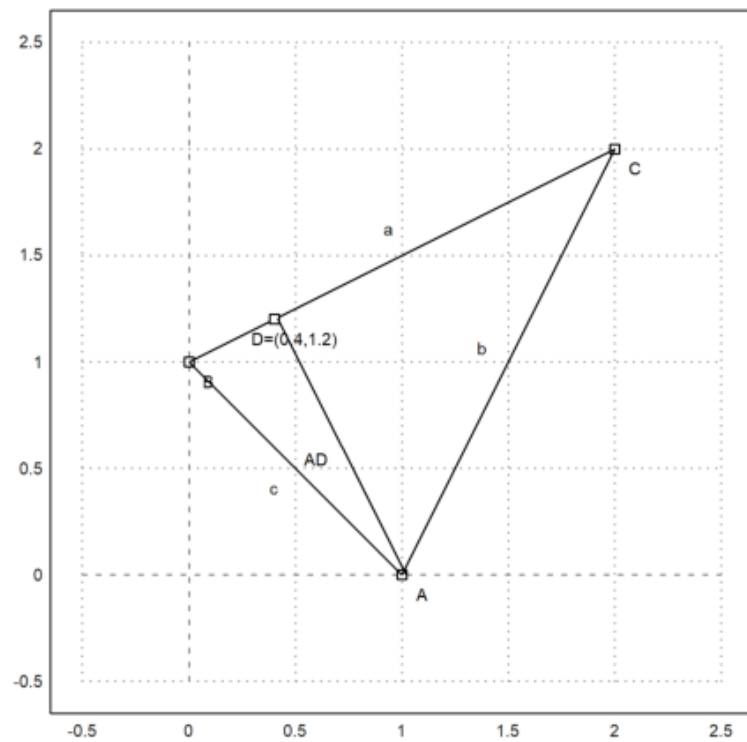
Setelah itu kita akan mencari koodinat titik D.

```
>plotPoint(D,value=1): // koordinat D ditampilkan
```





```
>aspect(1); plotSegment(A,D): // tampilkan semua gambar hasil plot...()
```



**Menghitung luas dari segitiga 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

norm(A-D):mencari panjang ruas garis AD

Atau kita dapat mencari luas segitiga tersebut dengan fungsi berikut

```
>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

## Menentukan Besar Sudut

---

Jika akan mencari sudut C dalam satuan derajat

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

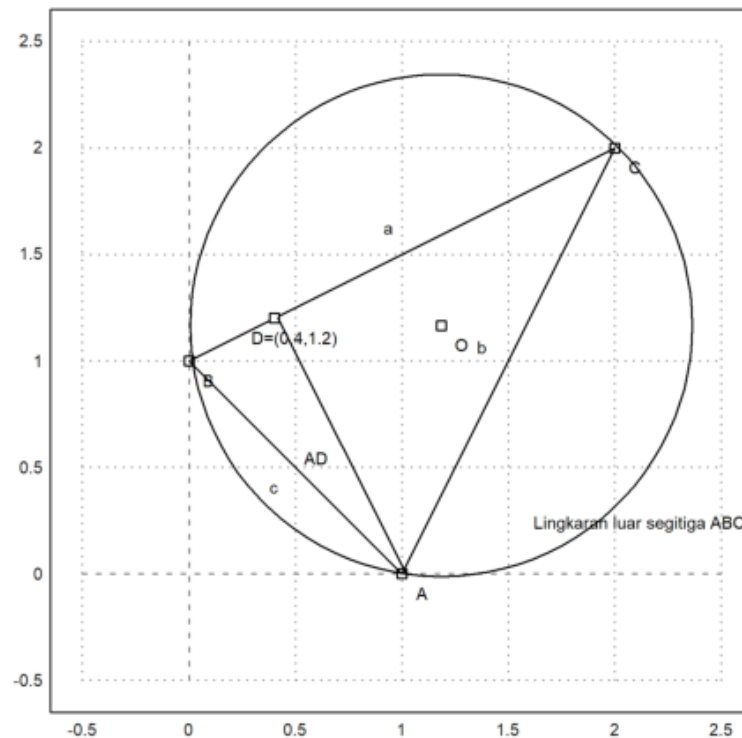
36°52'11.63''

Besar sudut C adalah 36° 52 menit dan 11.63 detik

## Menggambar Lingkaran Luar Segitiga

---

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



Dari gambar plot tersebut kita dapat menghitung koordinat titik pusat dan jari-jari lingkaran luar dengan perintah berikut

```
>O, R
```

```
[1.16667, 1.16667]
1.1751130198
```

## Menggambar Lingkaran Dalam Segitiga

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
```

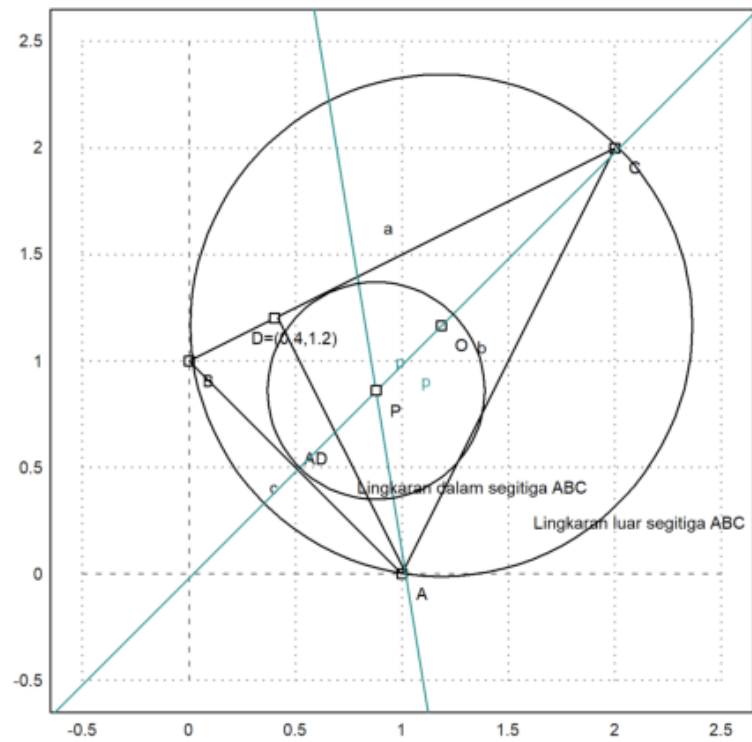
```
[0.86038, 0.86038]
```

Setelah didapat koordinat P, tambahkan semuanya ke 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
```

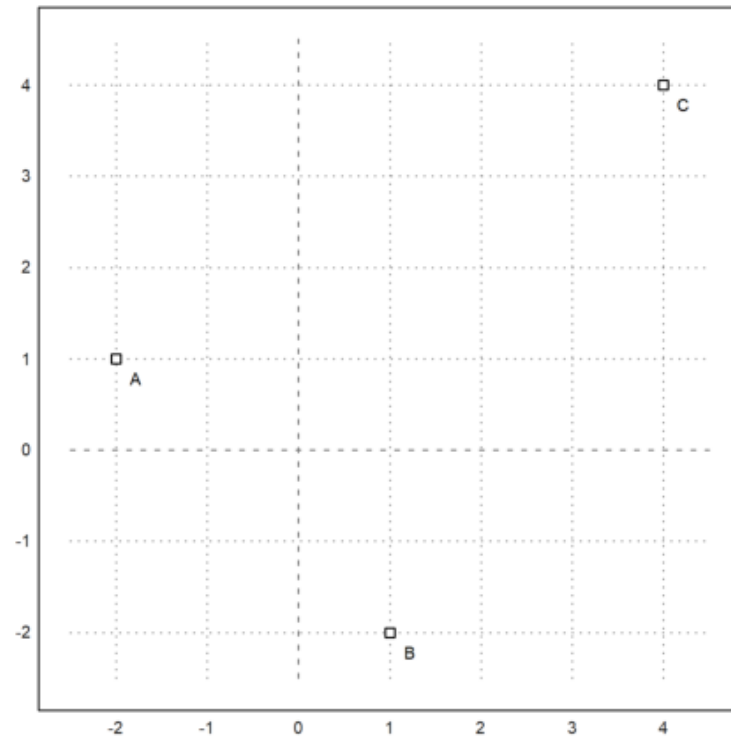


## Latihan

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

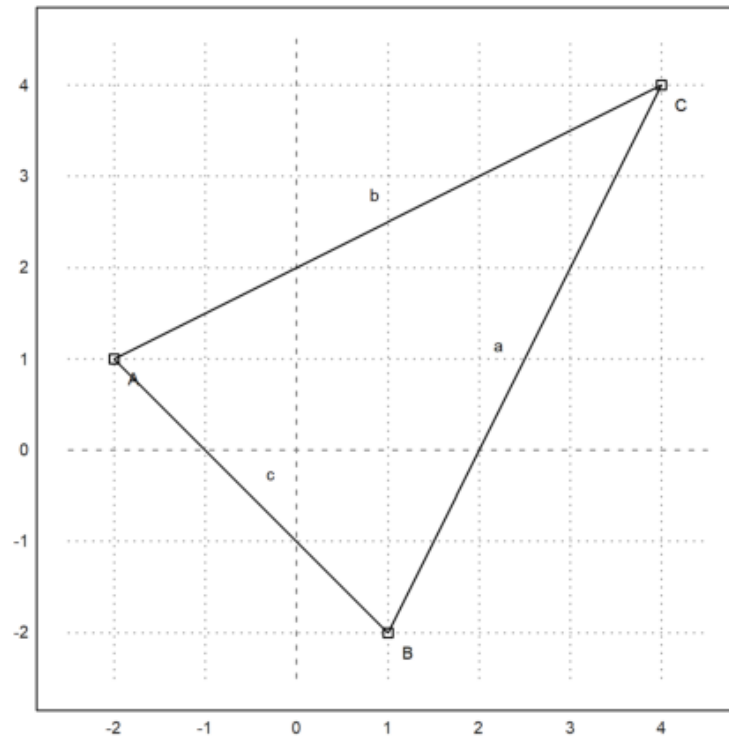
```
>setPlotRange(-2.5,4.5,-2.5,4.5);
>A=[-2,1]; plotPoint(A,"A");
>B=[1,-2]; plotPoint(B,"B");
>C=[4,4], plotPoint(C,"C"):
```

[4, 4]



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

```
>plotSegment(A,B,"c");  
>plotSegment(B,C,"a");  
>plotSegment(A,C,"b");  
>aspect(1):
```



Segitiga tersebut merupakan segitiga sama kaki

3. Hitung luas segitiga tersebut

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

13.5

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

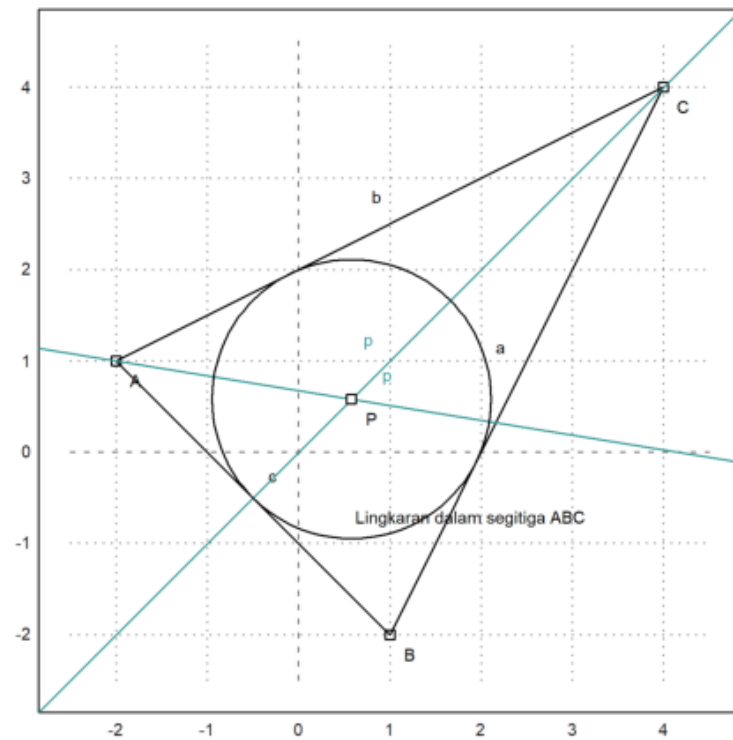
```
>l=angleBisector(A,C,B);
>g=angleBisector(C,A,B);
>P=lineIntersection(l,g)
```

[0.581139, 0.581139]

```
>color(5); plotLine(l); plotLine(g); color(1);
>plotPoint(P,"P");
>r=norm(P-projectToLine(P,lineThrough(A,B)))
```

1.52896119631

```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
```

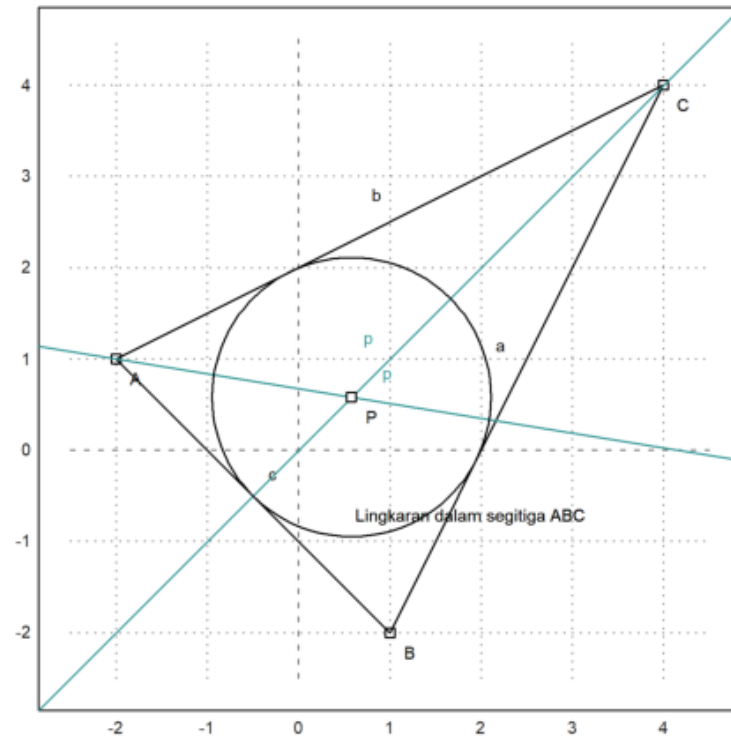


##### 5. Gambar jari-jari lingkaran dalam.

```
>r=norm(P-projectToLine(P,lineThrough(A,B)))
```

```
1.52896119631
```

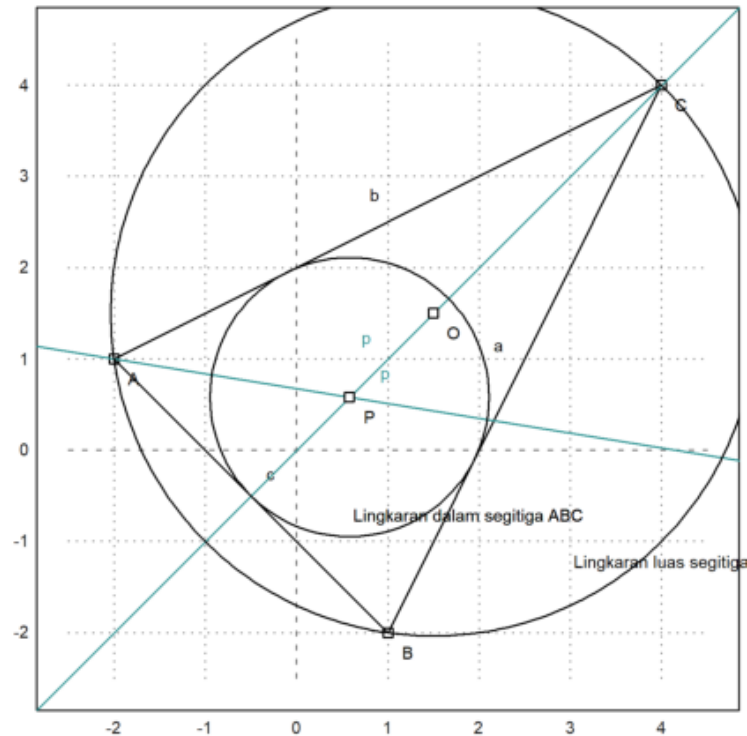
```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
```



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

```
>c=circleThrough(A,B,C);
>R=getCircleRadius(c);
>O=getCircleCenter(c);
>plotPoint(O,"O");
>plotCircle(c,"Lingkaran luar segitiga ABC");
```





```
>LL := pi*(R^2) // luas lingkaran luar
```

```
39.2699081699
```

```
>LD := pi*(r^2) // luas lingkaran dalam
```

```
7.34417132895
```

## Contoh 2: Persamaan Garis dan Koordinat Titik dengan Geometri

### Simbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima

Geometri simbolik adalah cabang geometri yang menggunakan simbol-simbol matematis dan aljabar untuk merepresentasikan dan memanipulasi objek-objek geometri, seperti titik, garis, bidang, dan bangun tiga dimensi.

File geometri menyediakan fungsi yang sama (dan lebih banyak lagi) dari Maxima. Namun, kita dapat menggunakan perhitungan simbolik sekarang.

### Menentukan persamaan garis melalui 2 titik

```
>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 perhitungannya simbolis.

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

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

## Mencari persamaan garis menggunakan metode simbolik

```
>$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)
```

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

Selanjutnya kita mencari garis yang tegak lurus dengan BC, dan melalui titik A

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

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

Hasil [2,1,2] merepresentasikan garis dengan persamaan

$$2x + y = 2$$

dimana persamaan tersebut tegak lurus dengan garis yang melewati titik B dan C serta melalui titik A

### Mencari Titik Potong Garis h dengan Garis BC

---

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

$$\begin{bmatrix} 2 & 6 \\ - & - \\ 5 & 5 \end{bmatrix}$$

Mencari proyeksi dari titik A ke titik D, maksudnya kita akan menarik garis dari titik A menuju titik D, dan menghasilkan koordinat yang sama

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

$$\begin{bmatrix} 2 & 6 \\ 5 & 5 \end{bmatrix}$$

### Mencari Panjang garis AD

---

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

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

### Menentukan Titik Pusat dan Jari-jari

---

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

$$\begin{bmatrix} 7 & 7 & 5 \\ 6 & 6 & 3\sqrt{2} \end{bmatrix}$$

Menampilkan nilai jari-jari dalam bentuk desimal

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

$$1.178511301977579$$

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

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

## Mencari Persamaan Garis Bagi dan Titik Potong Sudut ABC

---

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

$$y = x$$

Mencari titik potong 2 garis bagi tersebut

```
>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 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);
```

Perpotongan garis dengan lingkaran menghasilkan dua titik dan jumlah titik potong.

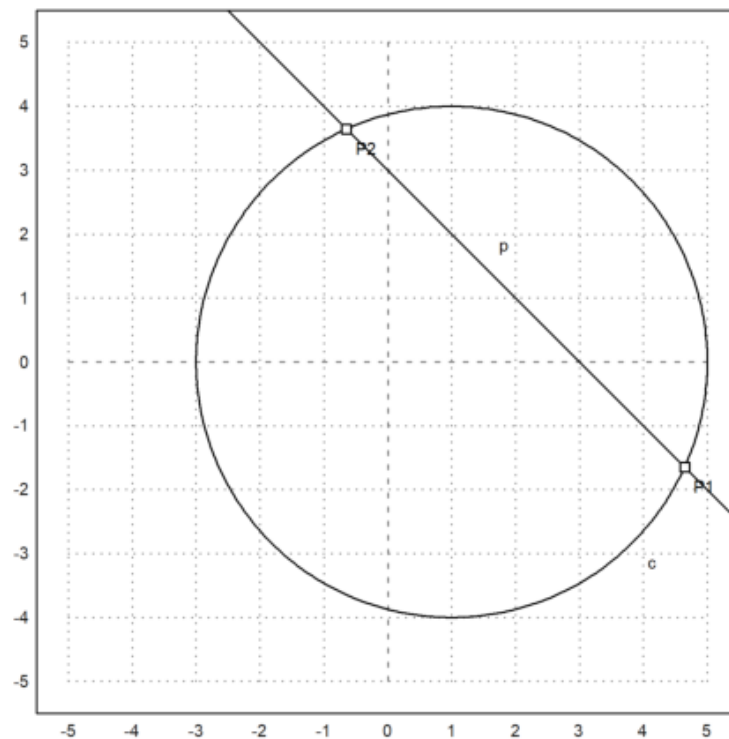
## Mencari Titik Koordinat P1 dan P2 dalam (x,y)

---

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

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

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



Begitu pula di Maxima.

Menampilkan titik pusat dan jari-jari

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

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

Menampilkan persamaan garis l yang melalui garis BC

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

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

Mencari titik potong antara sebuah garis l dan sebuah lingkaran c

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

$$\left[ \left[ \sqrt{7} + 2, 1 - \sqrt{7} \right], \left[ 2 - \sqrt{7}, \sqrt{7} + 1 \right] \right]$$

Menyederhanakan hasil solusi menggunakan radcan, terutama jika hasil melibatkan akar kuadrat

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

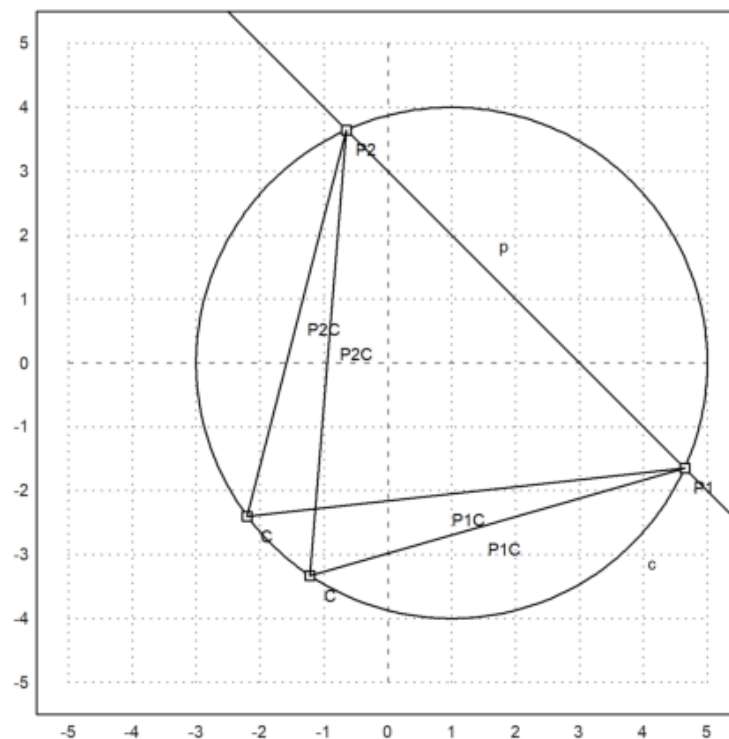
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
>degprint(computeAngle(P1,C,P2))
```

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;
```



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.

Mendefinisikan titik A dan titik B

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

Membuat lingkaran C1 dan C2

```
>c1=circleWithCenter(A,distance(A,B));  
>c2=circleWithCenter(B,distance(A,B));
```

---

## Mencari Titik Potong Antara Dua Lingkaran

---

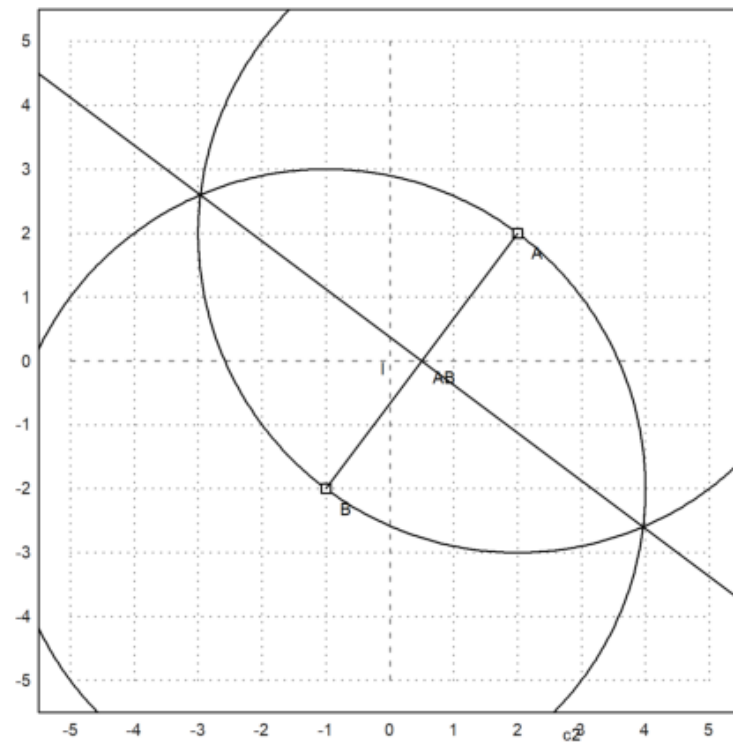
```
>{P1,P2,f}=circleCircleIntersections(c1,c2);
```

Membuat garis yang melalui P1 dan P2

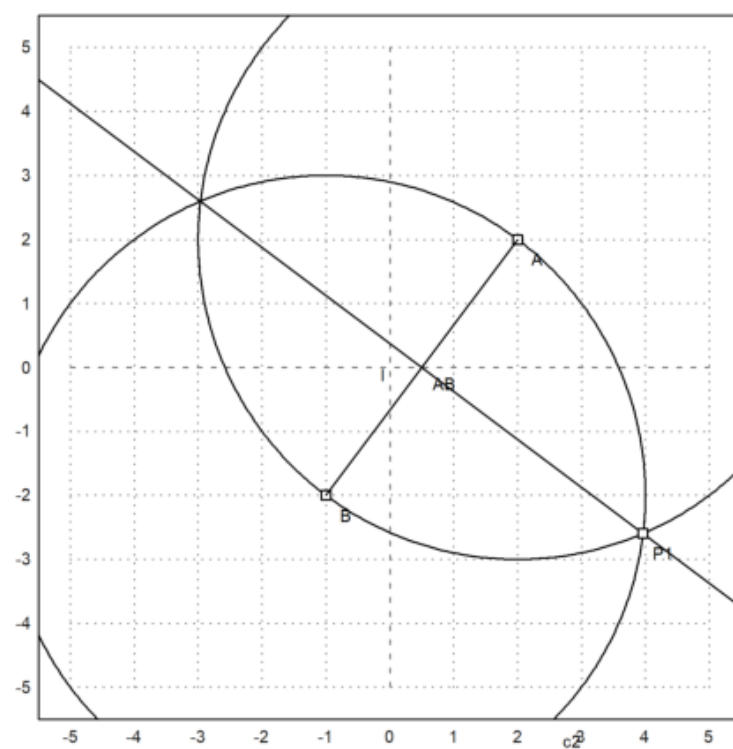
```
>l=lineThrough(P1,P2);
```

Mengatur rentang, titik, dan memplot lingkaran

```
>setPlotRange(5); plotCircle(c1); plotCircle(c2);  
>plotPoint(A); plotPoint(B); plotSegment(A,B); plotLine(l,"l");
```

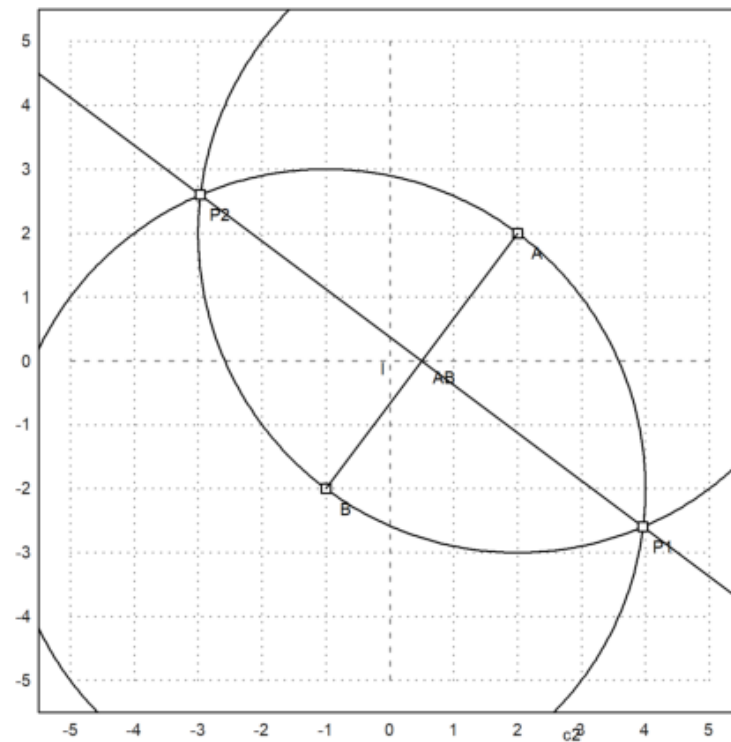


```
>plotPoint(P1,"P1"):
```





```
>plotPoint (P2, "P2") :
```



```
>P1, P2
```

```
[3.9641, -2.59808]
[-2.9641, 2.59808]
```

Selanjutnya akan dilakukan hal yang sama di Maxima dengan koordinat secara umum  
Membuat lingkaran dan mencari titik potong

```
>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 dapat disederhanakan dengan memecah y.

Langkah pertama, menentukan persamaan garis yang melalui dua titik potong P1 dan P2

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

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

Persamaan ini biasanya berbentuk linier,  $y=mx+by$ , dengan  $m$  sebagai gradien

Selanjutnya adalah menghitung persamaan garis tegak lurus tengah dari segmen yang menghubungkan A dan B. Garis tegak lurus tengah adalah garis yang membagi segmen AB secara tegak lurus

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

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

Selanjutnya kita akan menentukan persamaan garis yang melalui titik A dan B

```
>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:

$$m_1.m_2 = \frac{-(b_1 - a_1)}{(b_2 - a_2)} \times \frac{(b_2 - a_2)}{(b_1 - a_1)} = -1.$$

Artinya kedua garis tegak lurus.

### Contoh 3: Rumus Heron

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

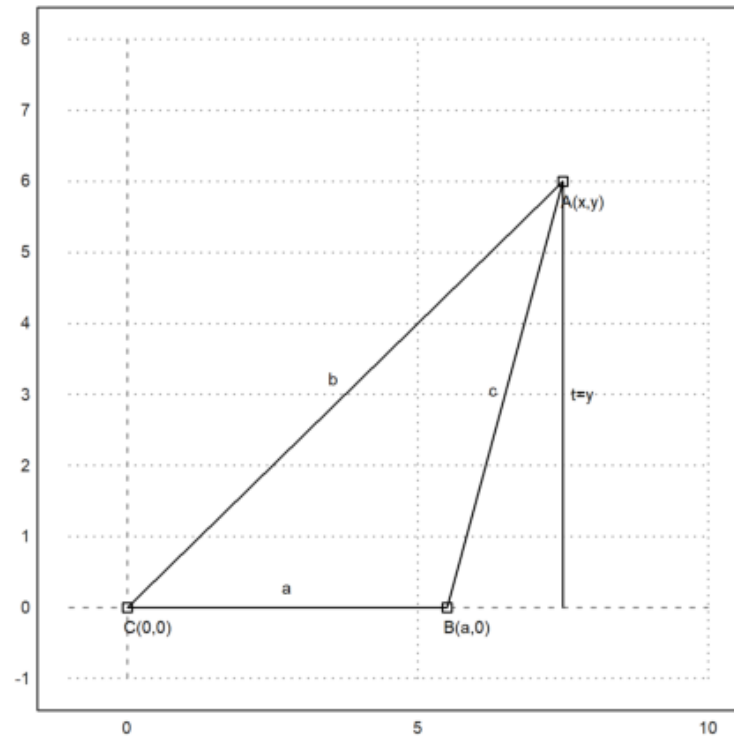
$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

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  $A(x,y)$ ,  $b=AC$ ,  $c=AB$ .

```
>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):
```



Luas segitiga ABC adalah

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

Nilai y didapat dengan langkah-langkah berikut

$$AC = \sqrt{(x-0)^2 + (y-0)^2}$$

$$b = \sqrt{(x)^2 + (y)^2}$$

$$b^2 = x^2 + y^2$$

atau

$$AB = \sqrt{(x-a)^2 + (y-0)^2}$$

$$c = \sqrt{(x-a)^2 + (y)^2}$$

$$c^2 = (x-a)^2 + y^2$$

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

$$-c^2 + b^2 + a^2$$

$$\left[ \begin{aligned} x &= \frac{-c^2 + 2b^2 + c^2 + 2a^2 - b^2 + 2a^2 - a^2}{2a} \\ y &= \frac{\sqrt{(-c^2 + 2b^2 + c^2 + 2a^2 - b^2 + 2a^2 - a^2)}}{2a} \end{aligned} \right]$$

$$\left[ \begin{aligned} x &= \frac{-c^2 + b^2 + a^2}{2a} \\ y &= \frac{\sqrt{(-c^2 + 2b^2 + c^2 + 2a^2 - b^2 + 2a^2 - a^2)}}{2a} \end{aligned} \right]$$

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

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

Setelah mendapat solusi, ekstrak solusi y Kita telah mendapatkan nilai y, selanjutnya kita akan memasukkan nilai y ke dalam rumus luas segitiga

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

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

Terbukti bahwa rumus heron merupakan rumus luas segitiga jika diketahui ketiga sisinya

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

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

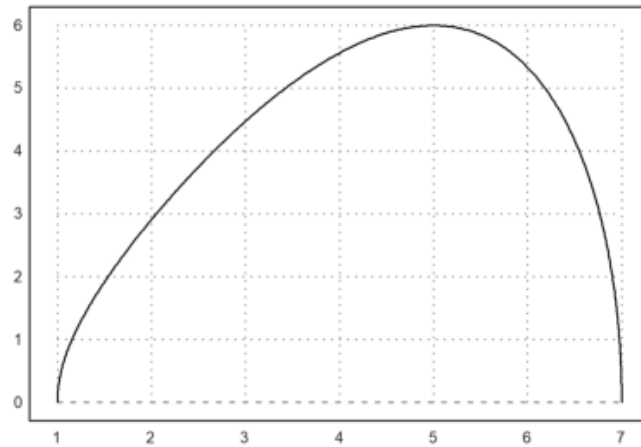
Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.

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

6

Dan juga terlihat 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 cari himpunan semua titik dimana  $b+c=d$  untuk semua konstan  $d$ . Hal ini dikenal dengan 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 buat fungsi dari ini

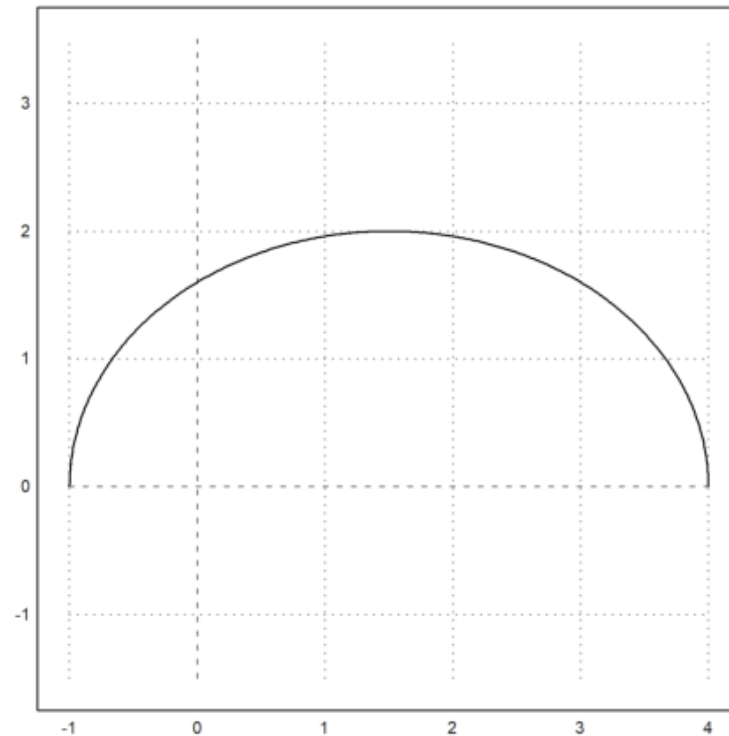
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

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

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

kita dapat menggambar himpunannya. Sisi  $b$  bervariasi dari 1 hingga 4. Terlihat jelas bahwa kita mendapatkan elips.

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



Kita dapat memeriksa persamaan umum untuk elips, yaitu:

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

dimana  $(x_m, y_m)$  adalah pusat, dan  $u$  serta  $v$  adalah setelah 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 lihat bahwa tinggi dan luas maksimal untuk  $x=0$ . Jadi, luas segitiga dengan  $a+b+c=d$  maksimal jika segitiga sama sisi. Kami ingin menurunkan ini secara analitik.

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

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

Kami mendapatkan beberapa minima, yang termasuk dalam segitiga dengan satu sisi 0 dan solusinya adalah  $a=b=c=d/3$ .

```
>$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

$$H(a, b, c)^2$$

terhadap  $a+b+c=d$

```
> 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])
```

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

Kita dapat memplot situasinya

Pertama-tama atur titik dengan Maxima

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

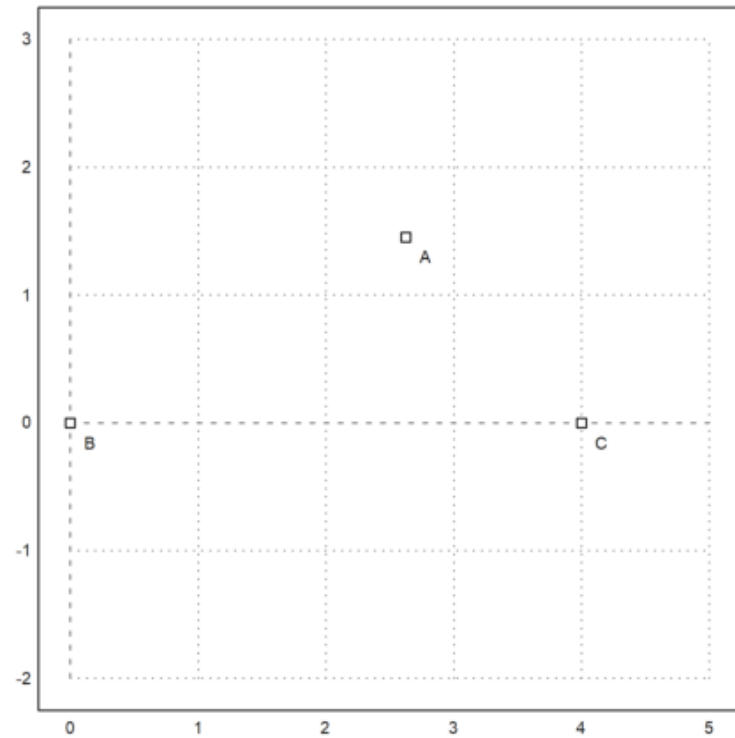
$$\left[ \frac{-c^2 + b^2 + a^2}{2a}, \frac{\sqrt{-c^4 + 2b^2c^2 + 2a^2c^2 - b^4 + 2a^2b^2 - a^4}}{2a} \right]$$

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

$$[a, 0]$$

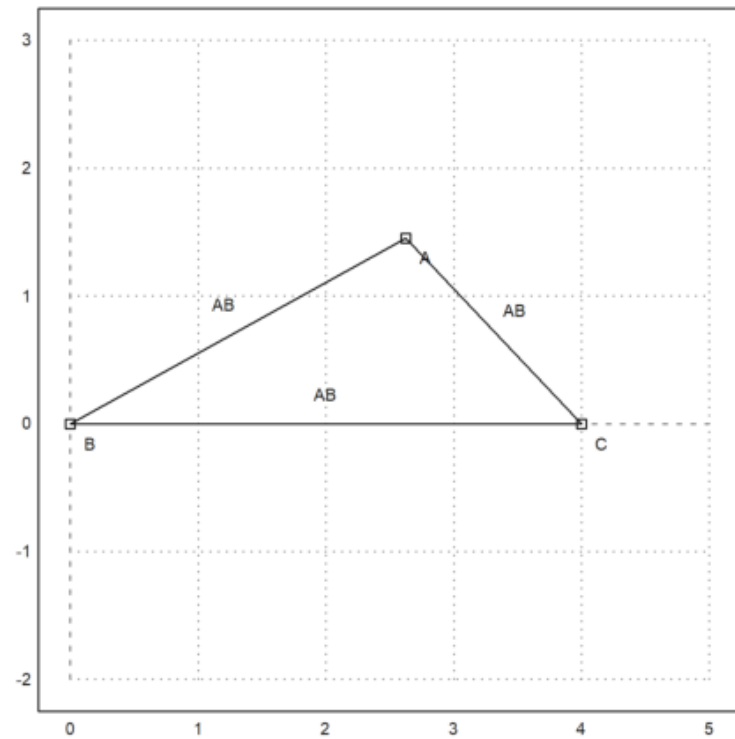
Lalu atur rentang plot, dan plot titik-titiknya

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



Plot garis segmennya juga

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





Cari garis tegak lurusnya dengan Maxima

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

Dan juga cari pusat lingkarannya.

```
>U &= lineIntersection(h,g);
```

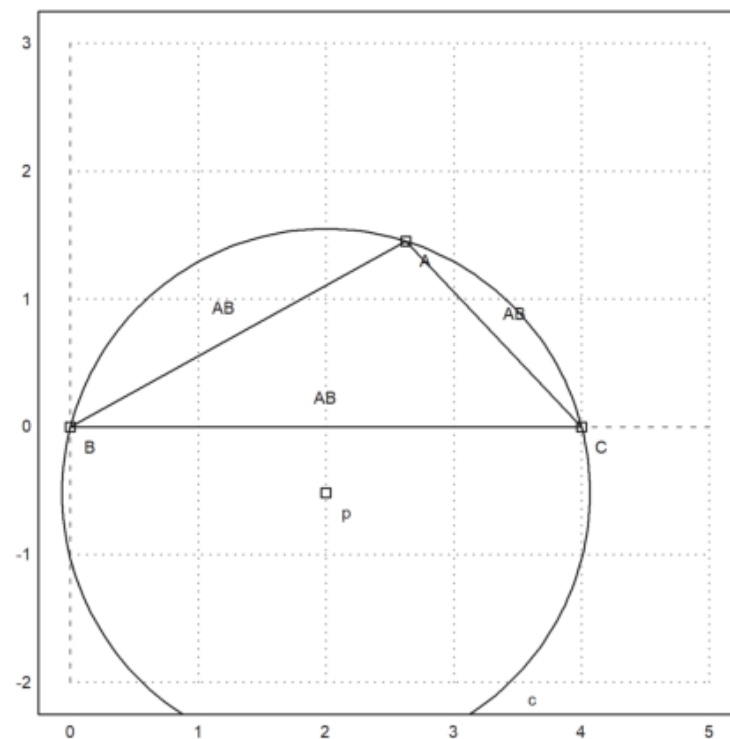
Kita mendapatkan rumus untuk jari-jari lingkaran

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

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

Let us add this to the plot.

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



Menggunakan geometri, kita mendapatkan rumus yang sederhana

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

untuk radiusnya. Kita dapat memeriksa apakah ini benar, dengan memanfaatkan Maxima. Maxima akan memfaktorkan ini hanya jika kita kuadratkan.

```
>$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 sebarang segitiga yang tidak sama sisi. Ini adalah garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk orthocenter, circumcenter, centroid, titik exeter dan pusat lingkaran sembilan titik segitiga.

- Orthocenter: titik potong garis tinggi
- Circumcenter: titik pusat lingkaran luar segitiga
- Centroid: titik berat atau pusat massa segitiga
- Titik exeter: titik istimewa yang terletak pada sumbu Euler

Pertama, mendefinisikan sudut-sudut segitiga yang akan digunakan dalam menghitung garis euler. Sudut-sudut ini akan diekspresikan dalam bentuk simbolik atau aljabar, sehingga mempermudah perhitungan lebih lanjut yang berkaitan dengan segitiga tersebut.

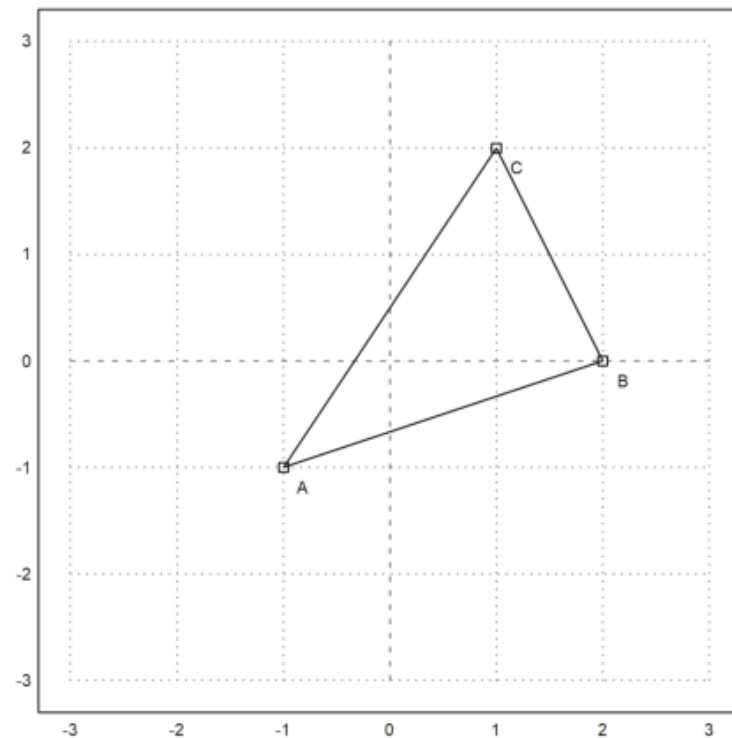
```
>A:=[-1,-1]; B:=[2,0]; C:=[1,2];
```

Untuk memplot objek geometri, kita menyiapkan area plot, dan menambah titik-titiknya. Semua plot objek geometri 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, dengan menggunakan formula determinan. Tentu saja kita harus mengambil nilai absolute dari hasilnya.

```
>$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 sebuah titik, sehingga titik tersebut berada di sisi positif dari bentuk Hesse. Memasukkan titik tersebut akan 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 akan menghitung keliling ABC

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

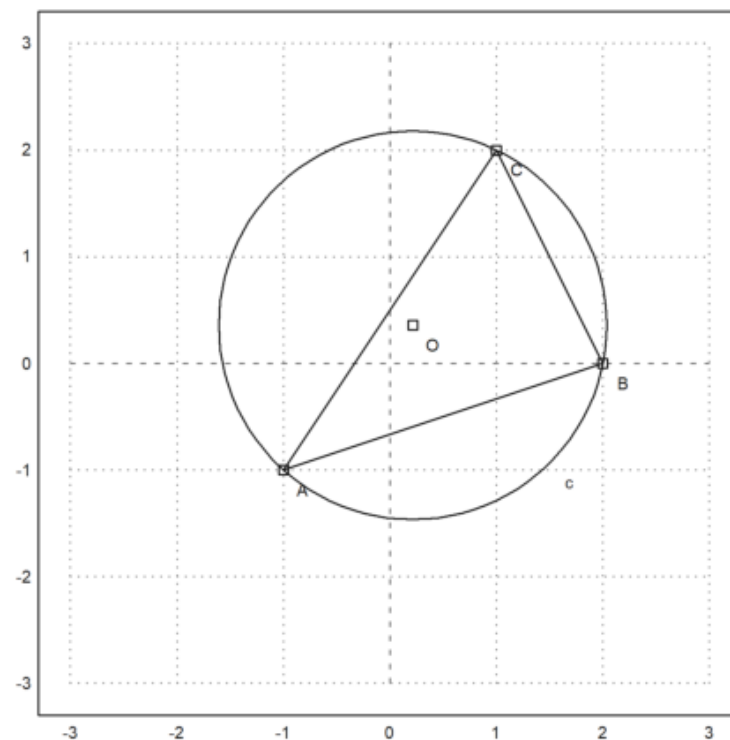
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

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

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

Plot lingkaran dan pusatnya. Cu dan U adalah simbolik. Kita akan mengevaluasi ekspresi ini untuk Euler

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



Kita dapat menghitung perpotongan ketinggian di ABC (pusat orthocenter) secara numerik dengan perintah berikut

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

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

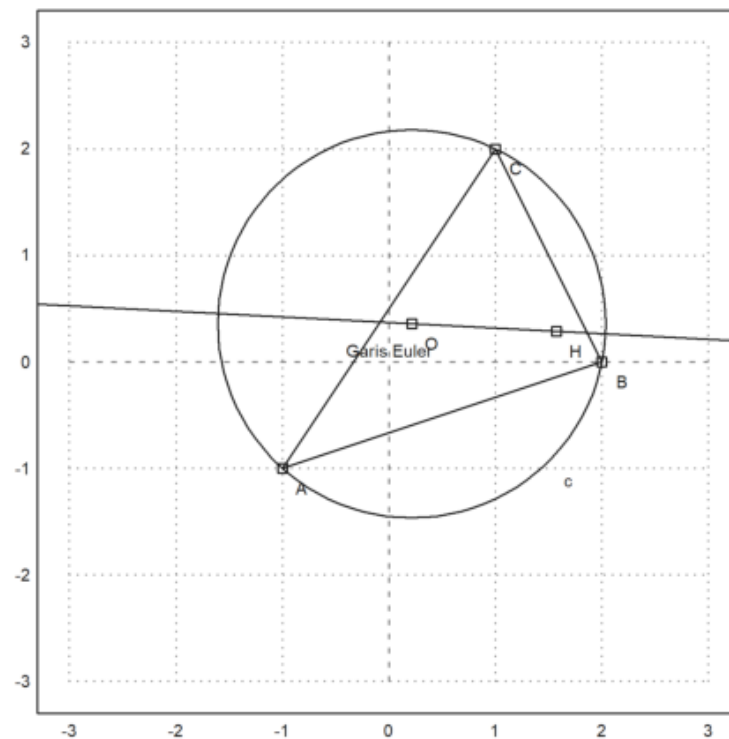
Sekarang kita dapat menghitung garis Euler dari segitiga tersebut

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

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

Tambahkan hal tersebut ke dalam plot kita

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

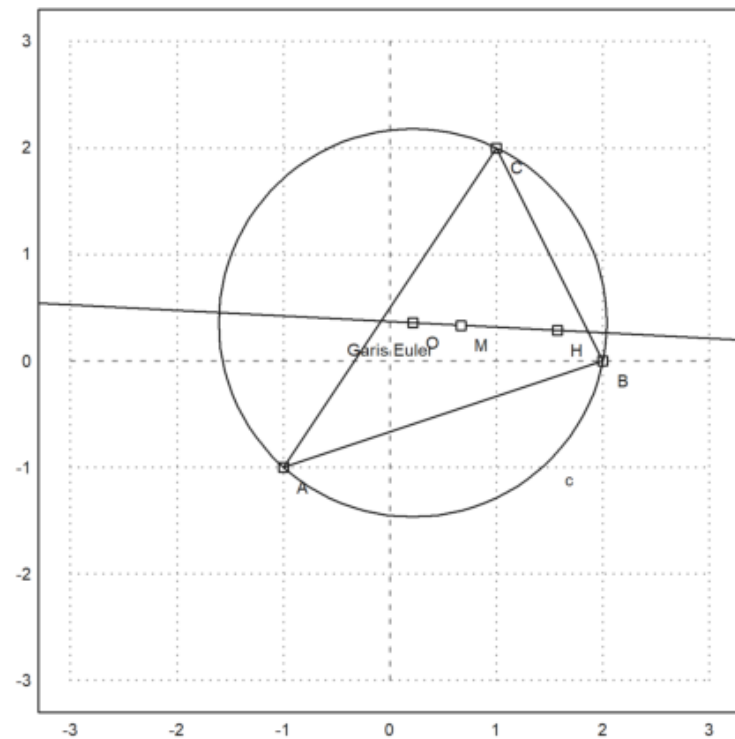


Pusat gravitasi harus berada pada garis ini

```
>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 mengatakan bahwa  $MH=2*MO$ . Kita perlu menyederhanakan radcan untuk mencapai hal ini

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

2

Fungsi ini juga mencakup fungsi untuk sudut

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

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

60°15'18.43''

Persamaan untuk bagian tengah lingkaran tidak terlalu 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 hitung juga ekspresi untuk jari-jari lingkaran yang tertulis

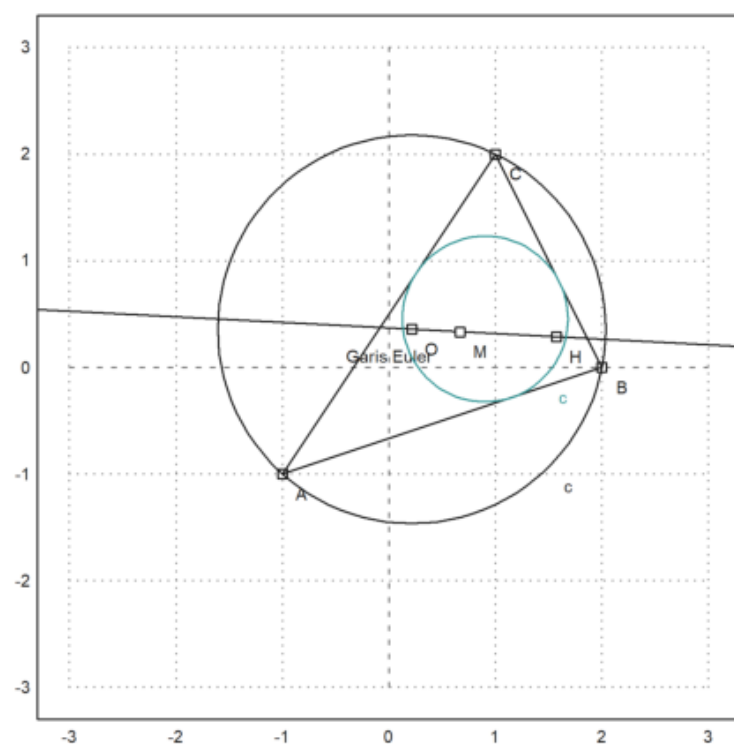
```
>r = distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
```

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

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

Lalu tambahkan ini ke dalam plot

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



## Parabola

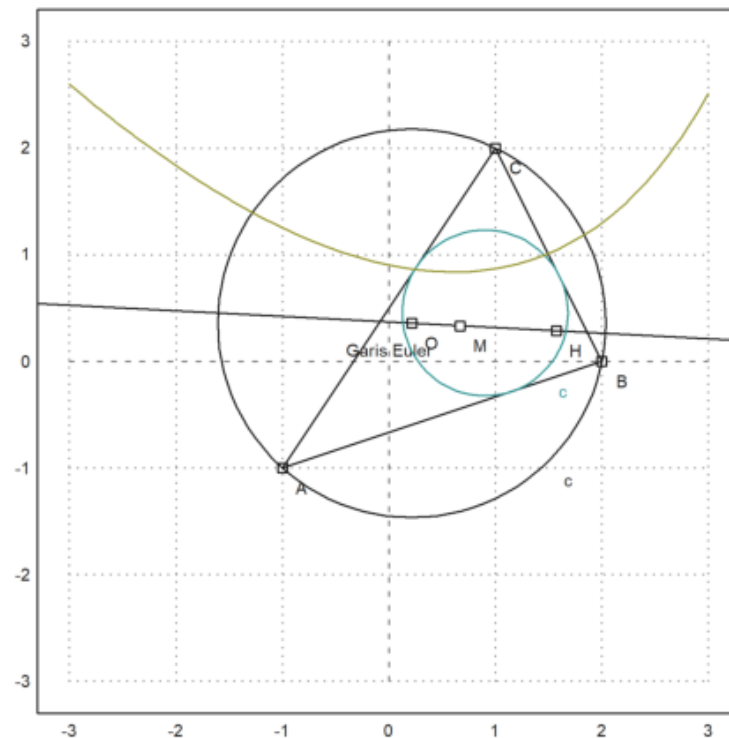
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Seharusnya ini menjadi beberapa fungsi, tetapi solver default dari Maxima dapat mencari solusi ini hanya jika kita mengkuadratkan persamaannya. Akibatnya, kita mendapatkan hasil yang palsu atau bukan sebenarnya.

```
>akar &= solve(getHesseForm(lineThrough(A,B),x,y,C)^2-distance([x,y],C)^2,y)
```

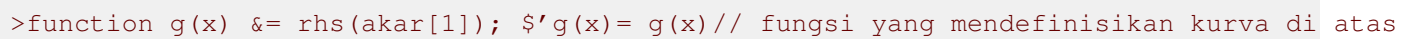
$$\begin{aligned} [y &= -3x - \sqrt{70} \sqrt{9-2x} + 26, \\ y &= -3x + \sqrt{70} \sqrt{9-2x} + 26] \end{aligned}$$

Solusi pertama adalah

$$y = -3x - \sqrt{70}\sqrt{9-2x} + 26$$



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



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

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

41

```
>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 ceramah N.J.Wildberger. Dalam buku "Devine PProportions", Wildberger bertujuan untuk menggantikan pengertian klasik dari jarak dan sudut dengan kuadran dan penyebaran. Menggunakan ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh, dan untuk tetap "rasional".

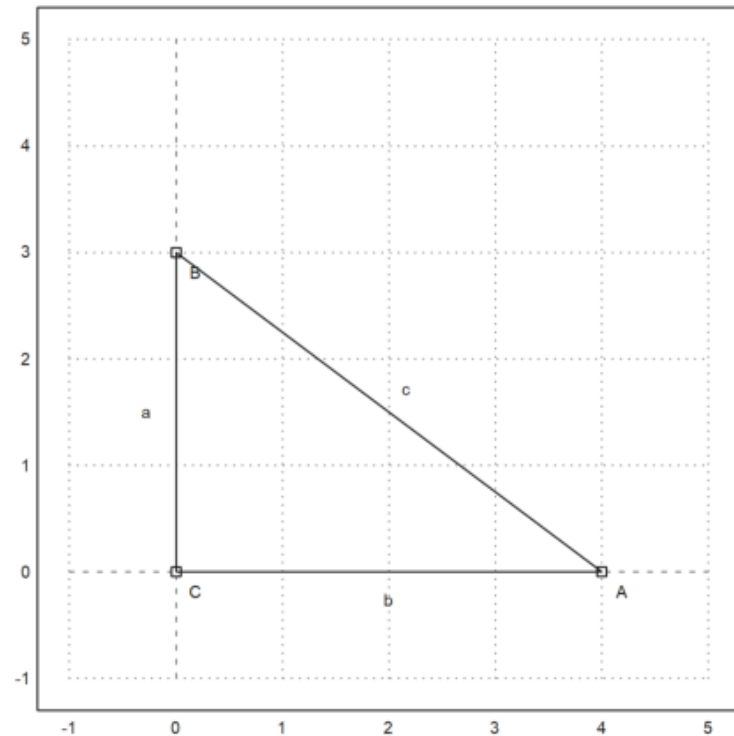
Dalam contoh berikut, Saya akan memperkenalkan konsep dan memecahkan beberapa masalah. Saya menggunakan komputasi simbolik Maxima disini, yang mana akan menyembunyikan keuntungan utama dari trigonometri rasional bahwa komputasi dapat dilakukan hanya dengan pensil dan kertas. Kalian dipersilahkan untuk mengecek hasilnya tanpa bantuan komputer.

Intinya adalah komputasi simbolik rasional biasanya menghasilkan hasil yang sederhana. Sebaliknya, trigonometri klasikal biasanya menghasilkan hasil trigonometri yang rumit, yang hanya mengevaluasi pendekatan numerik

```
>load geometry;
```

Untuk pengenalan pertama, kami menggunakan segitiga persegi panjang dengan proporsi Mesir yang terkenal yaitu 3,4, dan 5. Perintah berikut adalah perintah Euler untuk memplot bidang geometri yang berada di 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,

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

adalah tempat dimana  $w_a$  adalah sudut di A. Cara yang biasa digunakan untuk menghitung sudut ini adalah dengan mengambil invers dari fungsi sinus. Hasilnya adalah sudut yang tidak dapat dicerna, yang hanya dapat digambarkan perkiraannya.

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

36°52'11.63''

Trigonometri rasional mencoba untuk menghindari hal ini.

Definisi pertama dari trigonometri rasional adalah kuadran, yang menggantikan jarak. Faktanya, hal itu hanyalah jarak yang dikuadratkan. Berikut  $a$ ,  $b$ , dan  $c$  menotasikan kuadran dari sisi-sisinya.

Teorema Pythagoras menjadi  $a+b=c$

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

$$25 = 25$$

Definisi kedua dari trigonometri rasional adalah penyebaran. Spread mengukur pembukaan antar garis. Hal ini adalah 0 jika garisnya sejajar dan 1 jika garisnya persegi panjang. Ini adalah kuadrat sinus dari sudut antara dua garis.

Penyebaran garis AB dan AC pada gambar di atas didefinisikan sebagai

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

dimana a dan c adalah kuadrat dari sembarang persegi panjang dengan salah satu sudut di A.

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

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

Tentu saja hal ini lebih mudah untuk dihitung daripada sudut. Tapi kalaupun kehilangan properti bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja kita dapat mengonversi pendekaran kita untuk sudut wa menjadi sprad, dan mencetaknya sebagai pecahan.

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

$$9/25$$

Hukum kosinus trigonometri klasik menerjemahkan ke dalam "hukum silang" berikut.

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

dengan a, b, dan c adalah kuadrat dari sisi-sisi segitiga, dan sa adalah penyebaran sudut A. Sisi a seperti biasanya, merupakan sisi yang bersebrangan dengan sudut A.

Hukum ini diimplementasikan dalam file geometry.e yang kita muat ke Euler tadi.

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

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

Dalam kasus kita, kita mendapatkan

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

$$1024 = 1024$$

Mari kita gunakan hukum silang untuk menemukan penyebaran ke A. Untuk melakukan hal ini, kita buat hukum silang untuk kuadrat a, b, dan c dan selesaikan dengan penyebaran sa yang tidak diketahui.

Kalian dapat melakukan ini secara manual dengan mudah, tetapi Saya menggunakan Maxima. Tentu saja kita mendapatkan hasil yang telah kita miliki.

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

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

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

Kita sudah tau ini. Definisi dari penyebaran adalah kasus spesial dari hukum silang atau cross law.

Kita juga dapat memecahkan hal ini untuk semua a, b, dan c. Hasilnya adalah formula yang menghitung penyebaran dari sudut segitiga yang diberi kuadrat dari ketiga sisinya.

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

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

Kita dapat membuat fungsi dari hasilnya. Fungsi seperti itu sudah didefinisikan dalam file geometry.e di Euler.

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

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

Sebagai contoh, kita menggunakannya untuk menghitung sudut dari segitiga dengan sisi

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

Hasilnya rasional, yang mana tidak mudah untuk didapatkan jika kita menggunakan trigonometri klasik.

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

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

Ini adalah sudut dalam derajat.

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

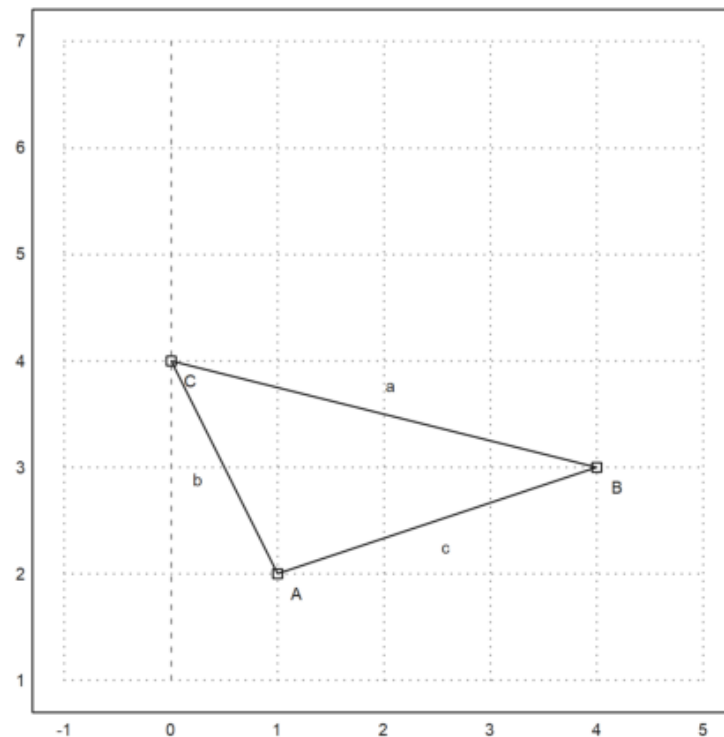
$$67^{\circ}47'32.44''$$

**Contoh Lain**

Sekarang, mari kita mencoba lebih banyak contoh yang lebih kompleks.

Kita mendefinisikan tiga sudut segitiga sebagai 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;
```



Dengan menggunakan Phytagoras, kita dapat dengan mudah mencari jarak dari dua titik. Saya pertama kali menggunakan fungsi jarak di Euler file untuk geometri. Fungsi jarak menggunakan geometri klasik.

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

$$\sqrt{10}$$

Euler juga memuat fungsi untuk kuadrat diantara dua titik.

Dalam contoh berikut, karena c+b bukan merupakan a, maka segitiganya bukan merupakan persegi panjang.

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

Pertama mari kita hitung sudut tradisional. Fungsi `computeAngle` menggunakan metode biasa berdasarkan hasil kali titik dua vektor. Hasilnya adalah beberapa pendekatan floating point.

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

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

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

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

```
>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 dapat melakukan hal yang sama dengan hukum silang. Kami memasukkan kuadran a, b, dan c ke dalam hukum silang dan menyelesaikan x.

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

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

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

Yaitu, apa yang dilakukan oleh penyebaran fungsi yang didefinisikan dalam "geometry.e".

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

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

Maxima mendapatkan hasil yang sama menggunakan trigonometri biasa, jika kita memaksanya. Itu menyelesaikan istilah  $\sin(\arccos(\dots))$  menjadi hasil pecahan. Sebagian besar siswa tidak dapat melakukan ini.

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

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

Setelah kita memiliki spread di B, kita dapat menghitung tinggi ha di sisi a. Ingat bahwa

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

Menurut definisi.

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

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

Gambar berikut telah dihasilkan dengan program geometri C.a.R., yang dapat menggambar kuadrat dan menyebar.

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

Menurut definisi, panjang ha adalah akar kuadrat dari kuadratnya.

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

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

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, bahwa kita berhadapan dengan kuadrat!

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

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

Rumus determinan biasa menghasilkan 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 spread di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat ("quadrea"?), faktorkan dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

Memang, 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

Kerugian dari spread adalah mereka tidak lagi hanya menambahkan sudut yang sama. Namun, tiga spread dari sebuah segitiga memenuhi aturan "triple spread" berikut.

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

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

Aturan ini berlaku untuk setiap tiga sudut yang menambah 180°.

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

Sejak menyebar

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

sama, aturan triple spread juga benar, jika

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

Karena penyebaran sudut negatif adalah sama, aturan penyebaran rangkap tiga juga berlaku, jika

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

Misalnya, kita dapat menghitung penyebaran sudut 60°. Ini 3/4. Persamaan memiliki solusi kedua, bagaimana-pun, di mana semua spread adalah 0.

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

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

Sebaran  $90^\circ$  jelas 1. Jika dua sudut dijumlahkan menjadi  $90^\circ$ , sebarannya menyelesaikan persamaan sebaran rangkap tiga dengan  $a, b, 1$ . Dengan perhitungan berikut kita mendapatkan  $a+b=1$ .

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

$$[x = 1 - y]$$

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

Jadi kita dapat menemukan penyebaran sudut berlipat ganda. Perhatikan bahwa ada dua solusi lagi. Kami membuat ini fungsi.

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

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

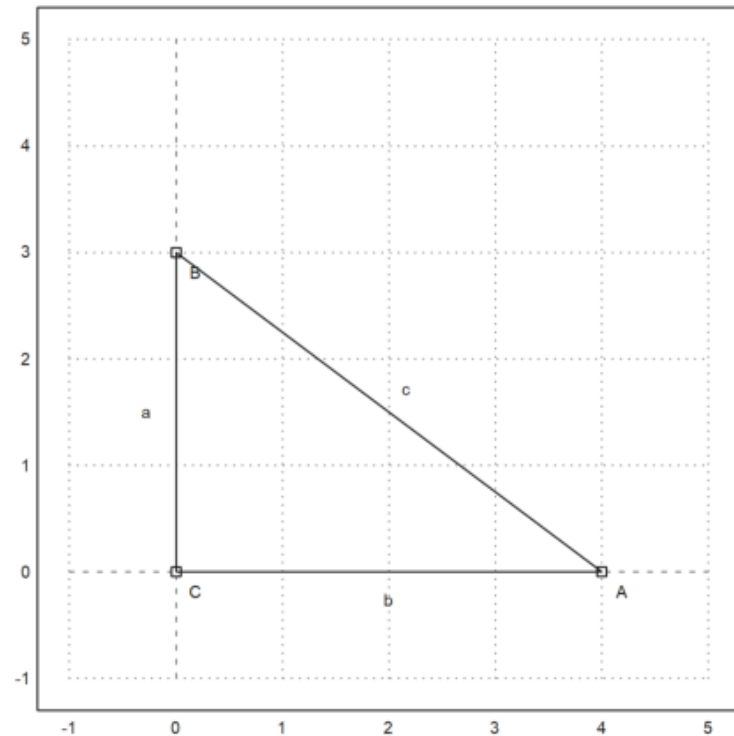
$$-4(a-1)a$$

## Sudut Pembagi

---

Ini situasinya, kita sudah tahu.

```
>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 garis bagi sudut di A. Tetapi kita ingin menyelesaikannya untuk umum  $a, b, c$ .

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

Jadi pertama-tama kita hitung penyebaran sudut yang dibagi dua di A, dengan menggunakan rumus sebaran rangkap tiga.

Masalah dengan rumus ini muncul lagi. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu pada sudut terbelah  $180^\circ$ -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 persegi panjang Mesir.

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

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

Kami dapat mencetak sudut dalam Euler, setelah mentransfer penyebaran ke radian.

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

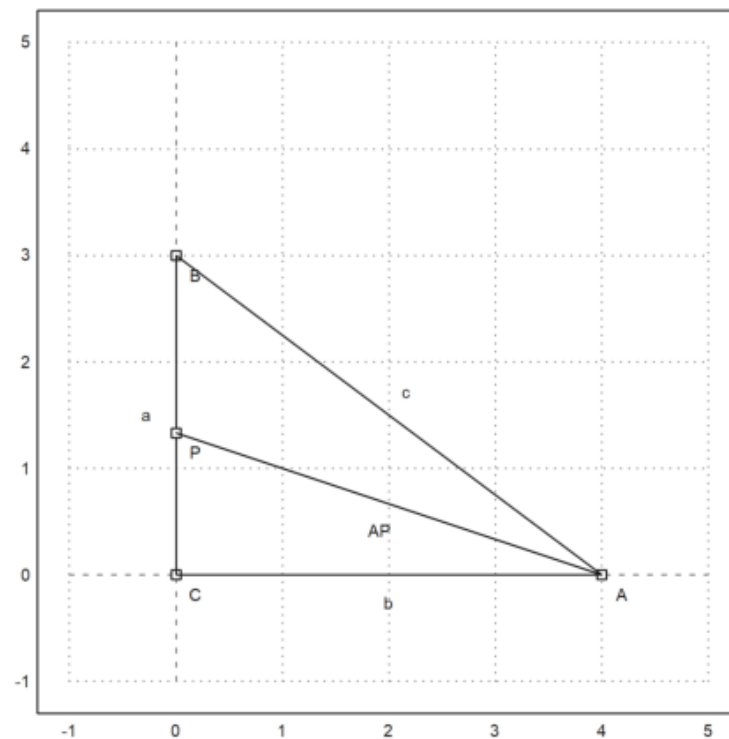
18°26'5.82''

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.

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

[0, 1.33333]

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



Mari kita periksa sudut dalam contoh spesifik kita.

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

0.321750554397

0.321750554397

Sekarang kita hitung panjang garis bagi AP.

Kami 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 dalam segitiga apa pun. Kuadratkan, itu diterjemahkan ke dalam apa yang disebut "hukum penyebaran"

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

di mana a,b,c menunjukkan quadrances.

Karena spread CPA adalah  $1-sa^2$ , kita dapatkan darinya  $bisa/1=b/(1-sa^2)$  dan dapat menghitung bisa (kuadrans dari garis-bagi sudut).

```
>&factor(ratsimp(b/(1-sa^2))); 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 dapat menghitung P menggunakan rumus spread.

```
>py&=factor(ratsimp(sa^2*bisa)); $py
```

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

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

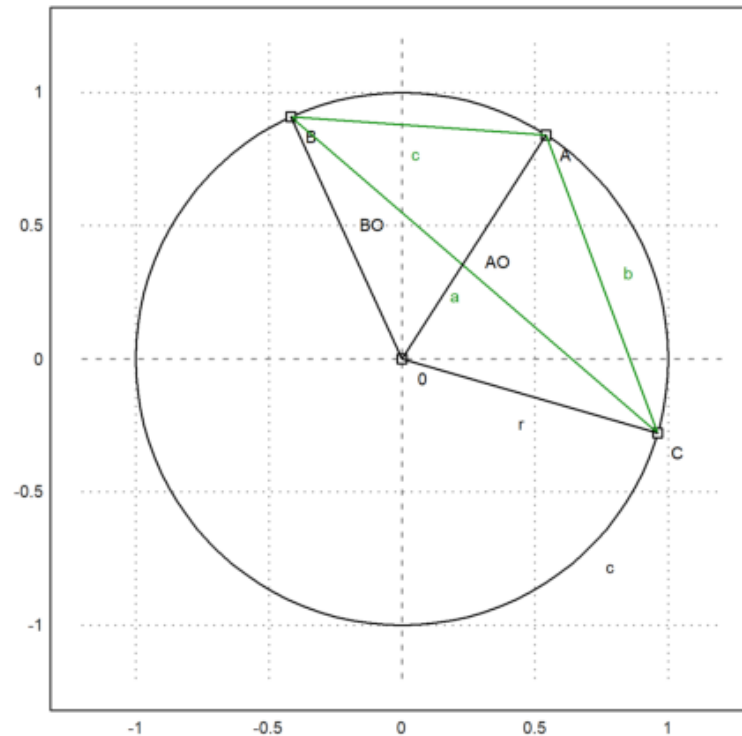
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

```
1.33333333333
```

## Sudut Akord

Perhatikan situasi berikut.

```
>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 dapat menggunakan Maxima untuk menyelesaikan rumus penyebaran rangkap tiga untuk sudut-sudut di pusat O untuk r. Jadi kita mendapatkan rumus untuk jari-jari kuadrat dari pericircle dalam hal kuadrat dari sisi.

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

```
>remvalue(a,b,c,r); // hapus nilai-nilai sebelumnya untuk perhitungan baru
```

Menentukan jari-jari keliling (circumradius) dari segitiga

```
>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 dapat menjadikannya sebagai fungsi Euler.

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

Mari kita periksa hasilnya untuk poin A,B,C.

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

Jari-jarinya memang 1.

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

1

Faktanya, spread CBA hanya bergantung 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 melihat bahwa sudut chord dari chord b adalah setengah dari sudut pusat.

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

0

## Contoh 6: Jarak Minimal pada Bidang

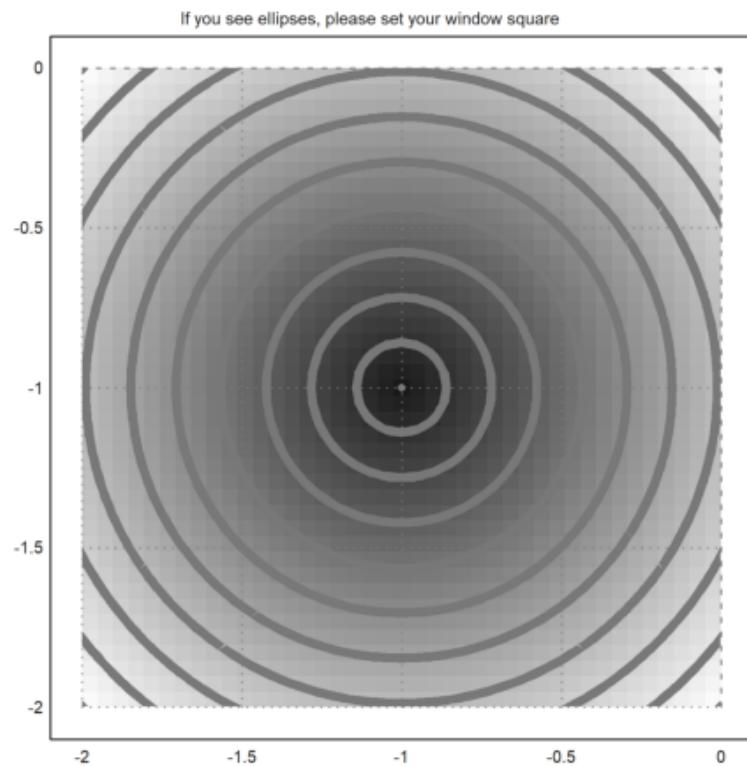
---

### Catatan awal

---

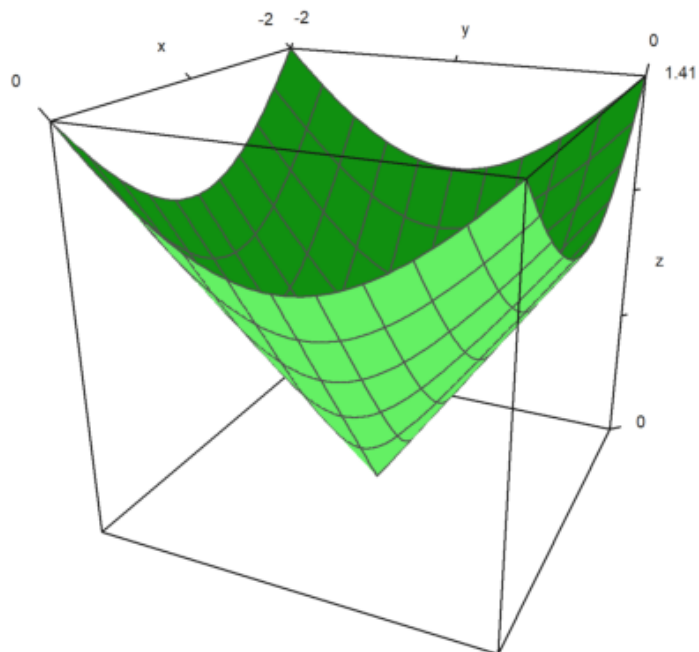
Fungsi yang, ke titik M di bidang, menetapkan jarak AM antara titik tetap A dan M, memiliki garis level yang agak sederhana: lingkaran berpusat di A.

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



dan grafiknya juga agak sederhana: bagian atas kerucut:

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



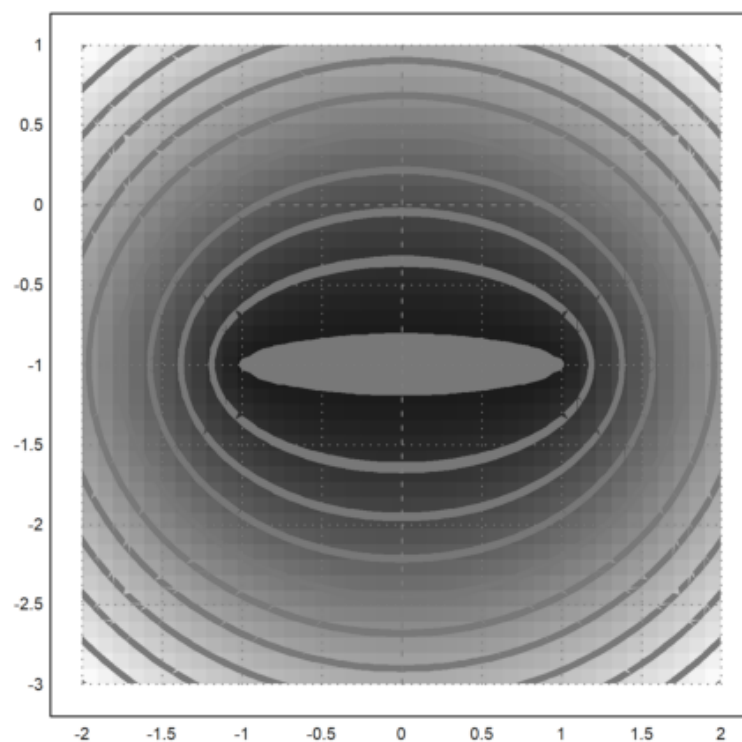


Tentu saja minimal 0 dicapai di A.

## Dua Titik

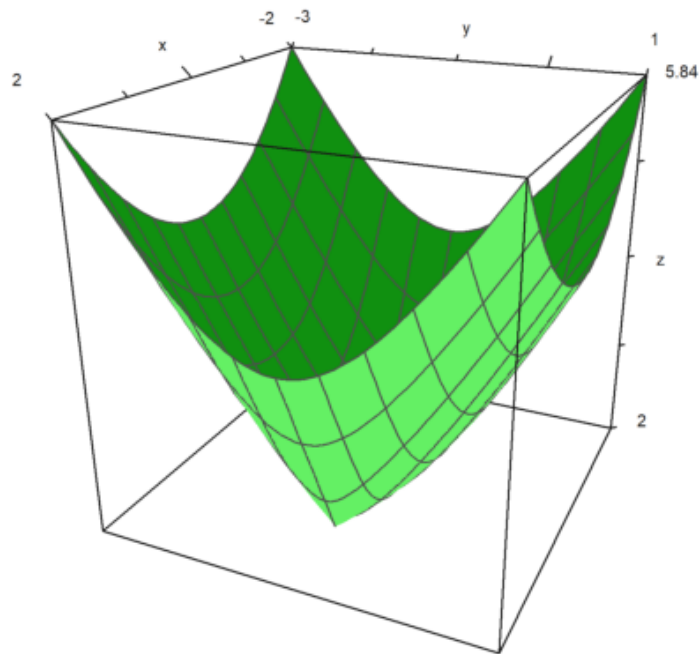
Sekarang kita lihat fungsi  $MA+MB$  dimana A dan B adalah dua titik (tetap). Ini adalah "fakta yang diketahui" bahwa kurva level adalah elips, titik fokusnya adalah A dan B; kecuali untuk AB minimum yang konstan pada segmen [AB]:

```
>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):
```



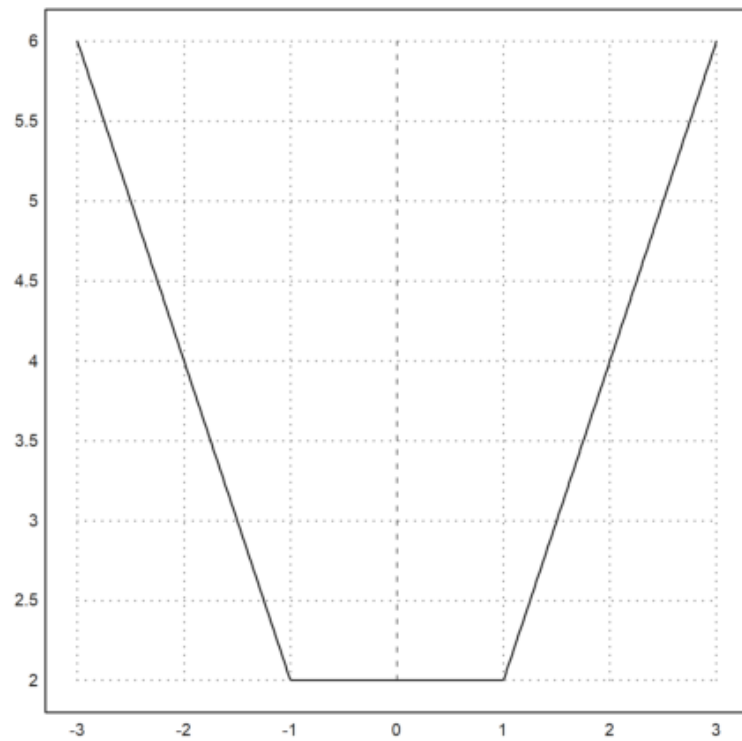
Grafiknya lebih menarik:

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



Pembatasan garis (AB) lebih terkenal:

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

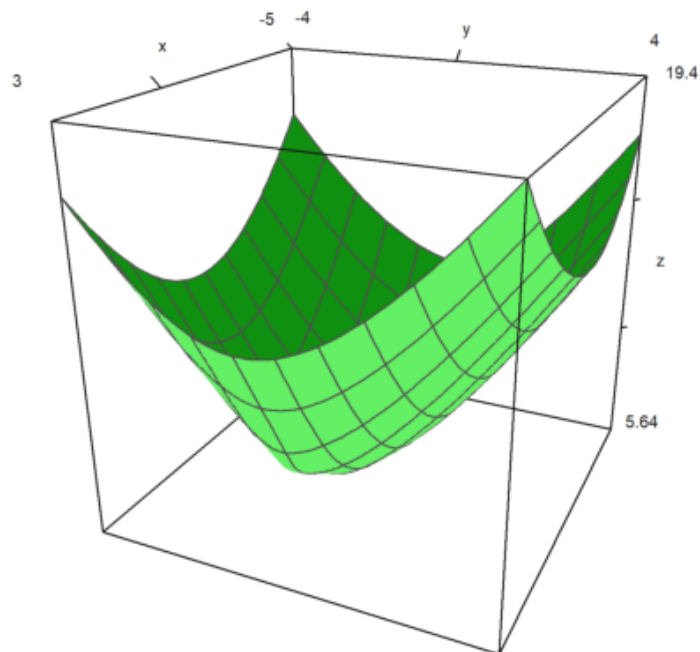


Sekarang hal-hal yang kurang sederhana: Ini sedikit kurang terkenal bahwa  $MA+MB+MC$  mencapai minimum pada satu titik pesawat tetapi untuk menentukan itu kurang sederhana:

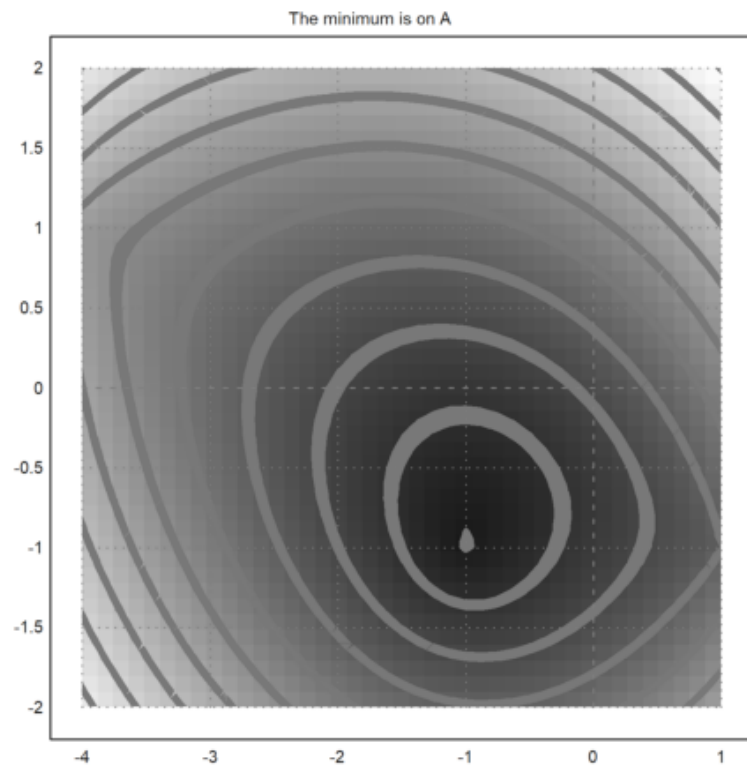
1) Jika salah satu sudut segitiga ABC lebih dari  $120^\circ$  (katakanlah di A), maka minimum dicapai pada titik ini (misalnya  $AB+AC$ ).

Contoh:

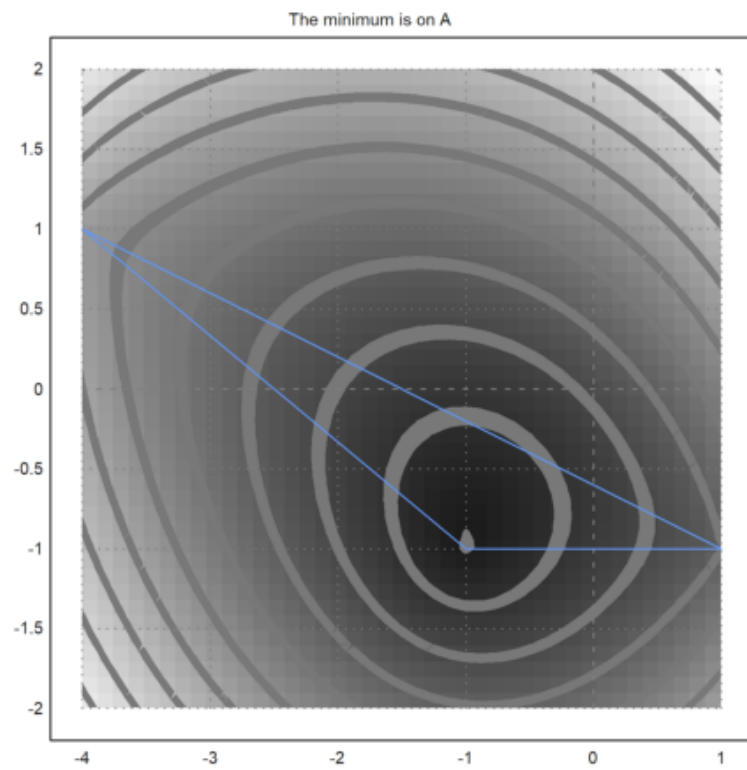
```
>C=[-4,1];  
>function d3(x,y):=d2(x,y)+sqrt((x-C[1])^2+(y-C[2])^2)  
>plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);  
>insimg;
```



```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A"):
```

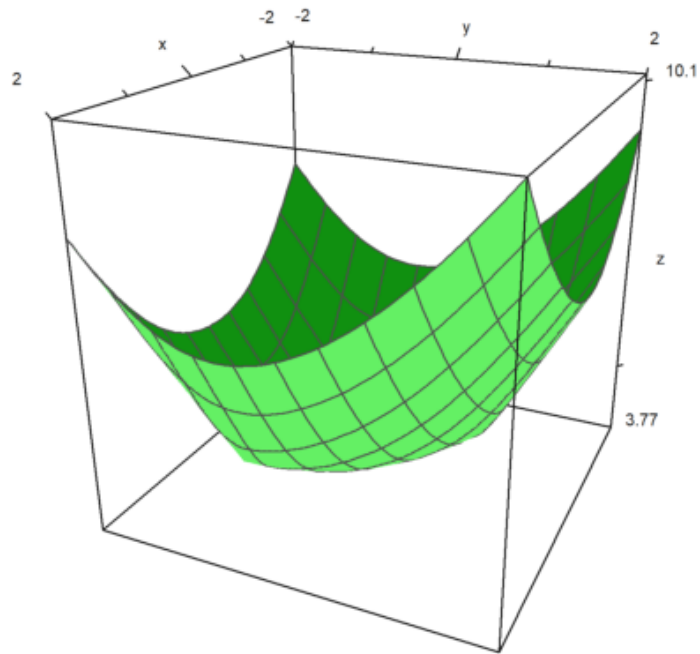


```
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

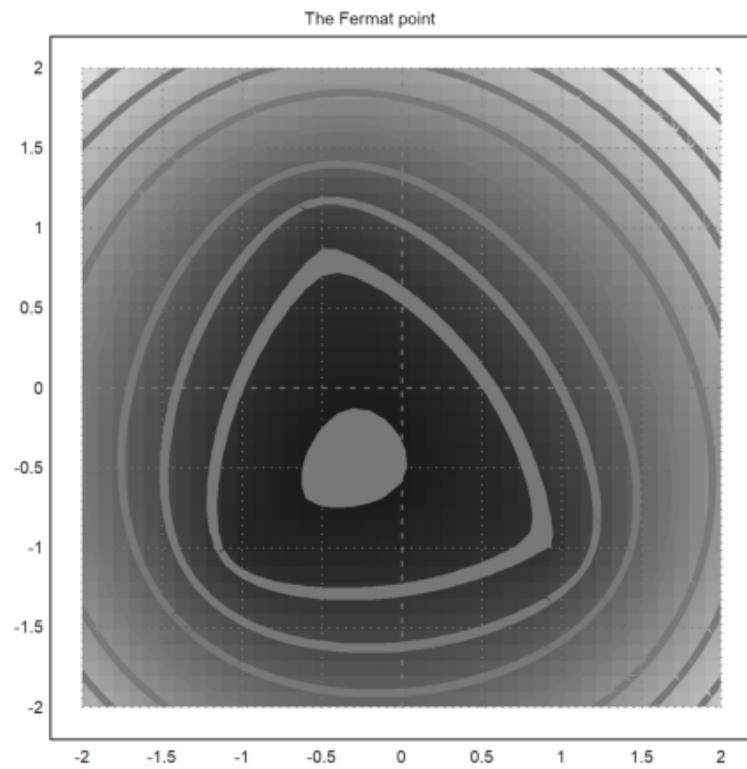


2) Tetapi jika semua sudut segitiga ABC kurang dari  $120^\circ$ , minimumnya adalah pada titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (maka masing-masing  $120^\circ$ ):

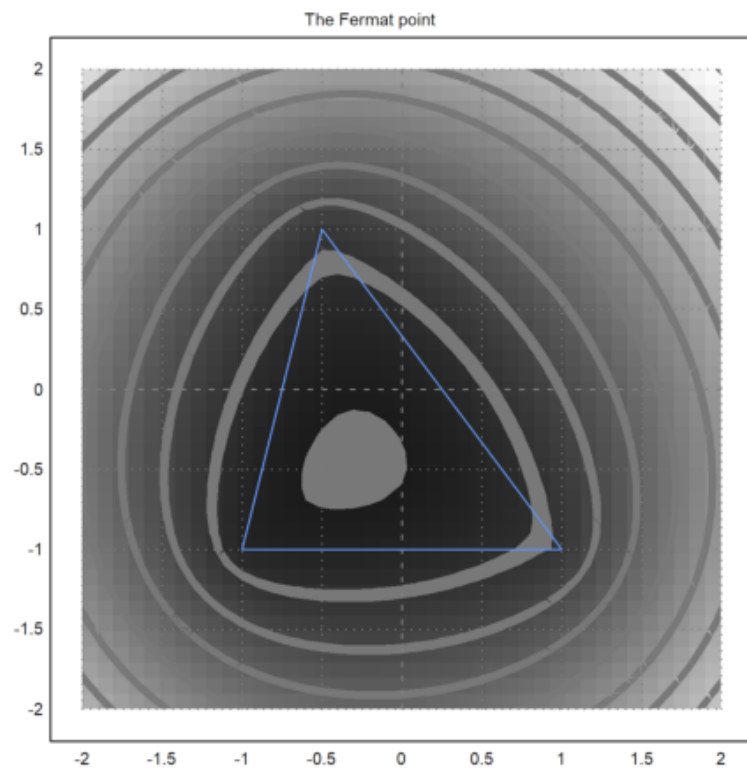
```
>C=[-0.5,1];  
>plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):
```



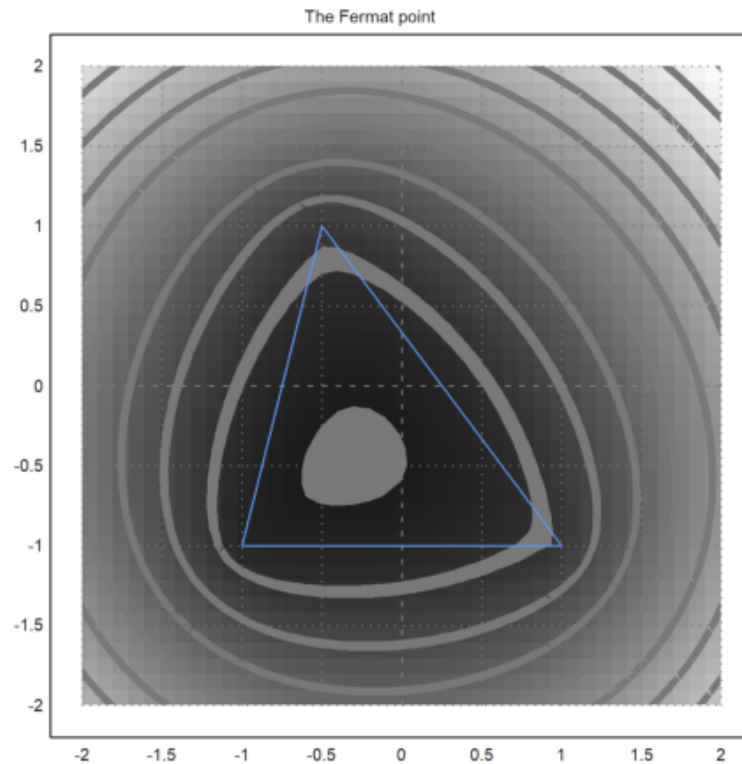
```
>fcontour("d3",xmin=-2,xmax=2,ymin=-2,ymax=2,hue=1,title="The Fermat point"):
```



```
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12):
```



```
>insimg;
```



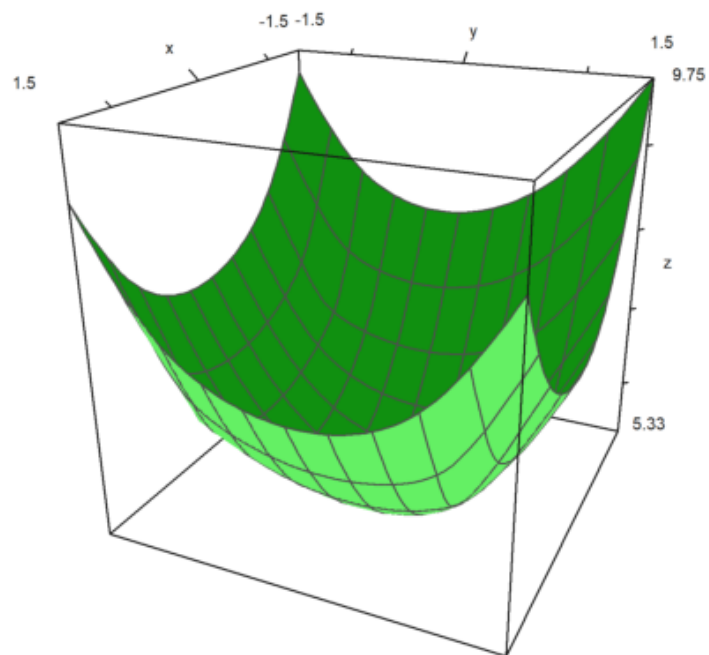
Merupakan kegiatan yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu soft yang ditulis di Jawa yang memiliki instruksi "garis kontur" ...

Semua ini di atas telah ditemukan oleh seorang hakim Perancis bernama Pierre de Fermat; dia menulis surat kepada dilettants lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di pajak penghasilan. Jadi titik unik F sedemikian rupa sehingga  $FA+FB+FC$  minimal, disebut titik Fermat segitiga. Tetapi tampaknya beberapa tahun sebelumnya, Torricelli Italia telah menemukan titik ini sebelum Fermat melakukannya! Bagaimanapun tradisinya adalah mencatat poin ini F...

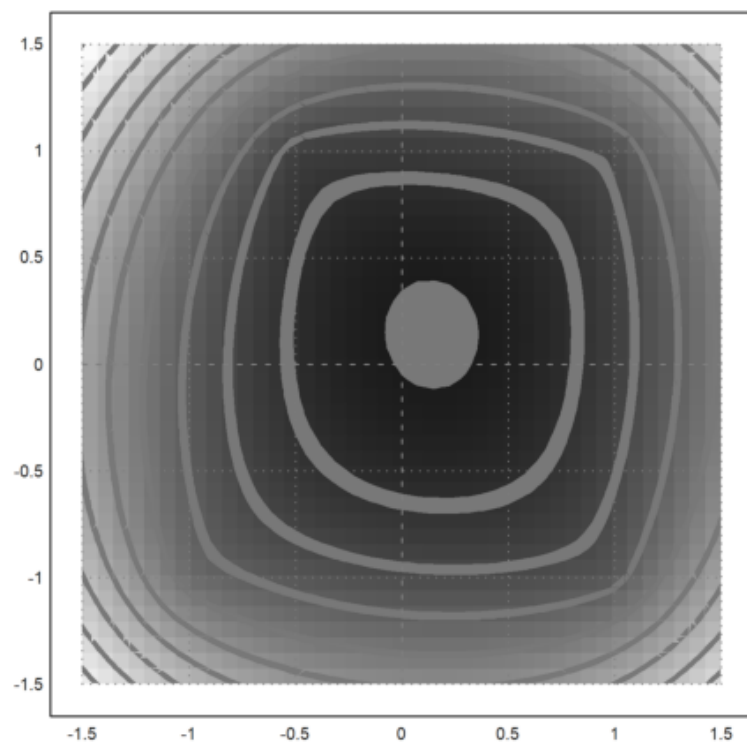
## Empat Titik

Langkah selanjutnya adalah menambahkan 4 titik D dan mencoba meminimalkan  $MA+MB+MC+MD$ ; katakan bahwa Anda adalah operator TV kabel dan ingin mencari di bidang mana Anda harus meletakkan antenna sehingga Anda dapat memberi makan empat desa dan menggunakan panjang kabel sesedikit mungkin!

```
>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):
```

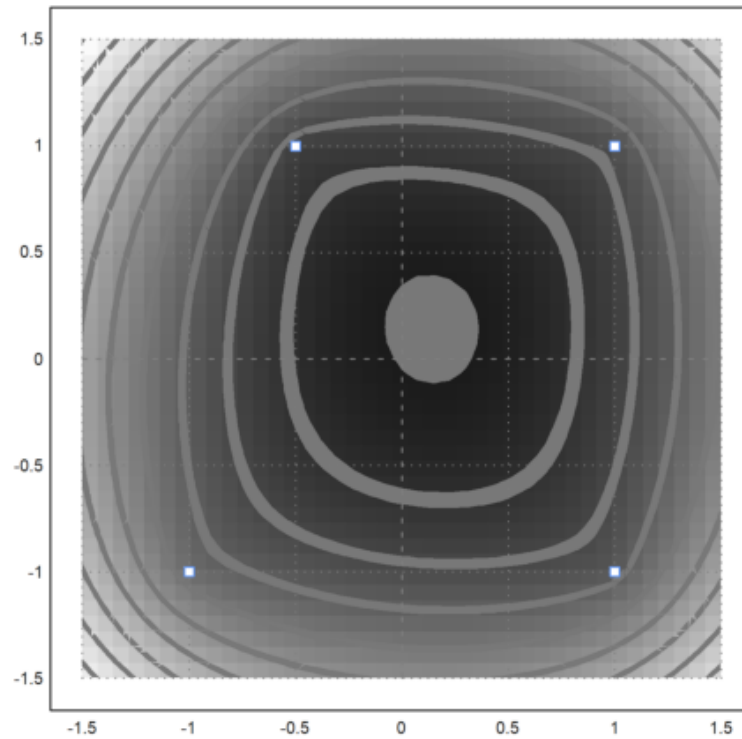


```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1):
```





```
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
```



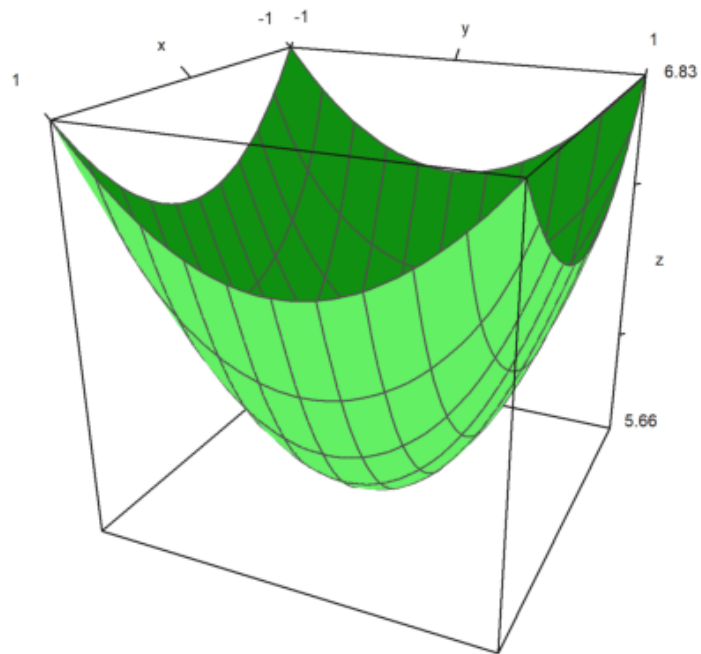
Masih ada minimum dan tidak tercapai di salah satu simpul A, B, C atau D:

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

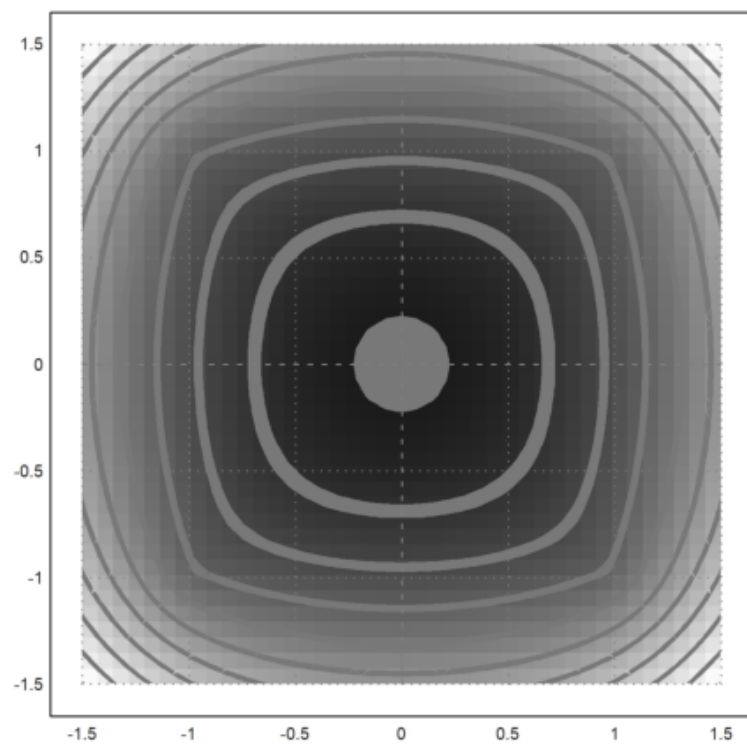
```
[0.142858, 0.142857]
```

Tampaknya dalam kasus ini, koordinat titik optimal adalah rasional atau mendekati rasional...  
Sekarang ABCD adalah persegi, kami berharap bahwa titik optimal akan menjadi pusat ABCD:

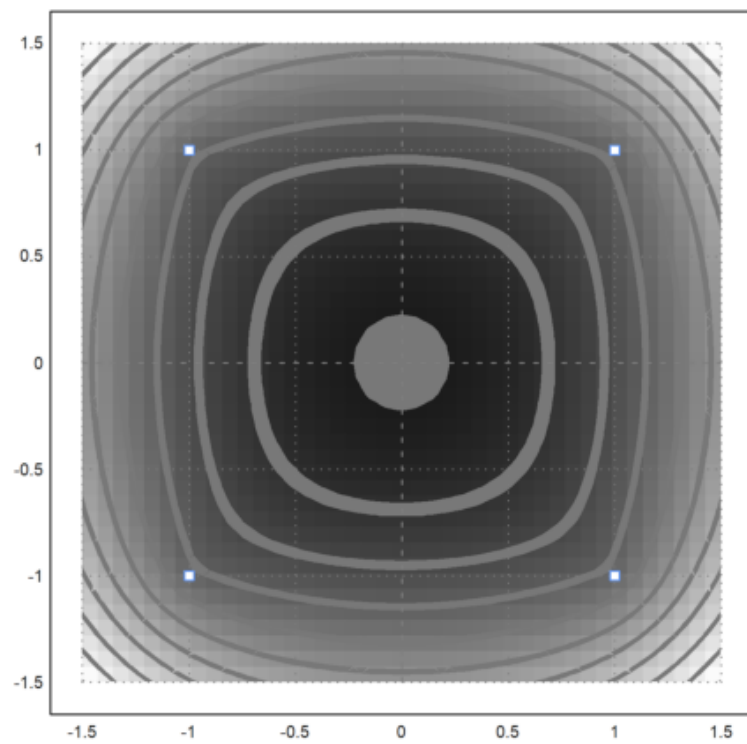
```
>C=[-1,1];
>plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
```



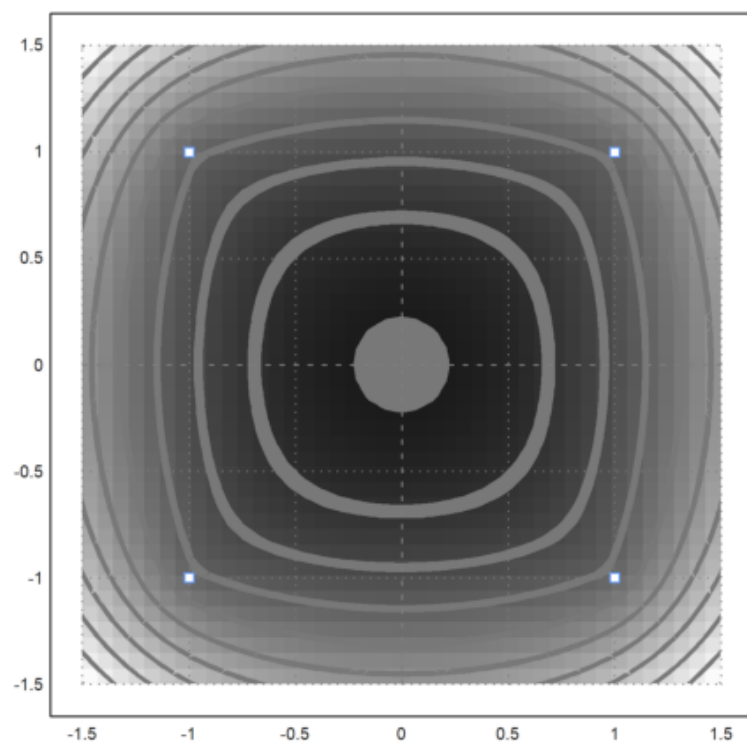
```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1):
```



```
>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 di jalur program.

Pertama kita hitung jari-jari bola.

Jika Anda melihat gambar di bawah, Anda melihat bahwa kita membutuhkan dua lingkaran yang menyentuh dua garis yang membentuk kerucut, dan satu garis yang membentuk bidang yang memotong kerucut.

Kami menggunakan file geometri.e dari Euler untuk ini.

```
>load geometry;
```

Pertama dua garis yang membentuk kerucut.

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

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

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

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

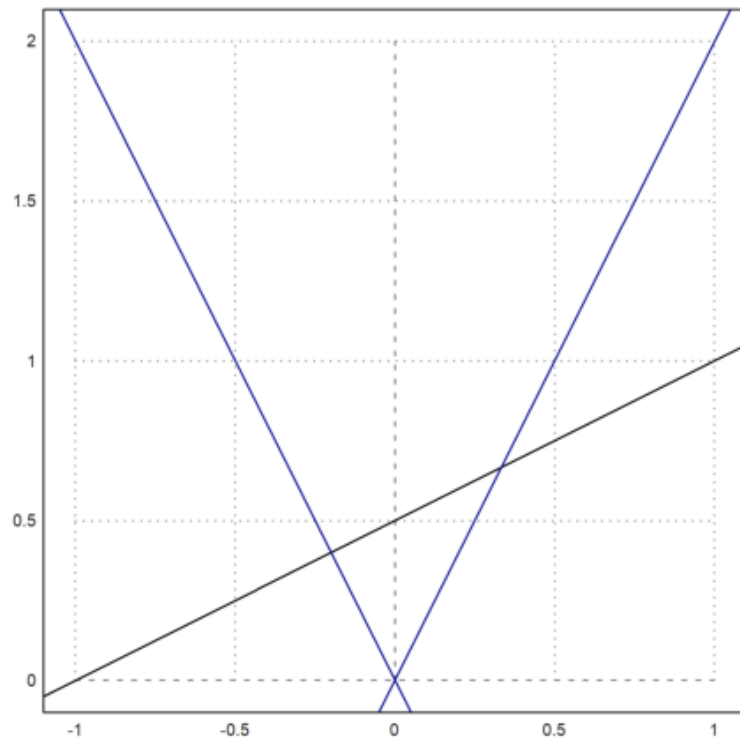
Kemudian baris ketiga.

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

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

Kita memplot semuanya sejauh ini

```
>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]
```

$[0, u]$

Hitung jarak ke gl

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

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

Hitung jarak ke g

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

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

Dan cari pusat dari kedua lingkaran dimana jaraknya sebanding.

```
>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.

Kami mengevaluasi solusi simbolis, dan menemukan kedua pusat, dan kedua jarak.

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

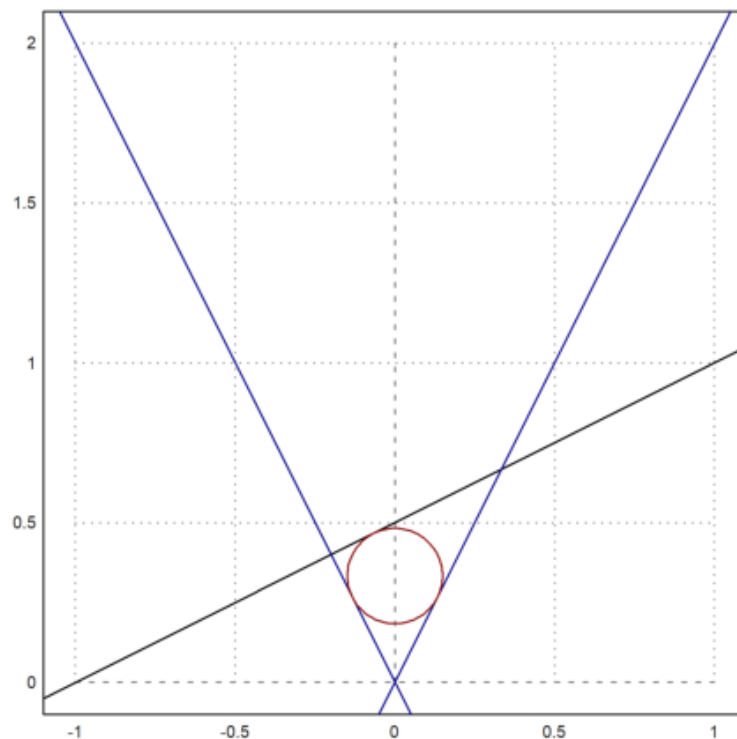
```
[0.333333, 1]
```

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

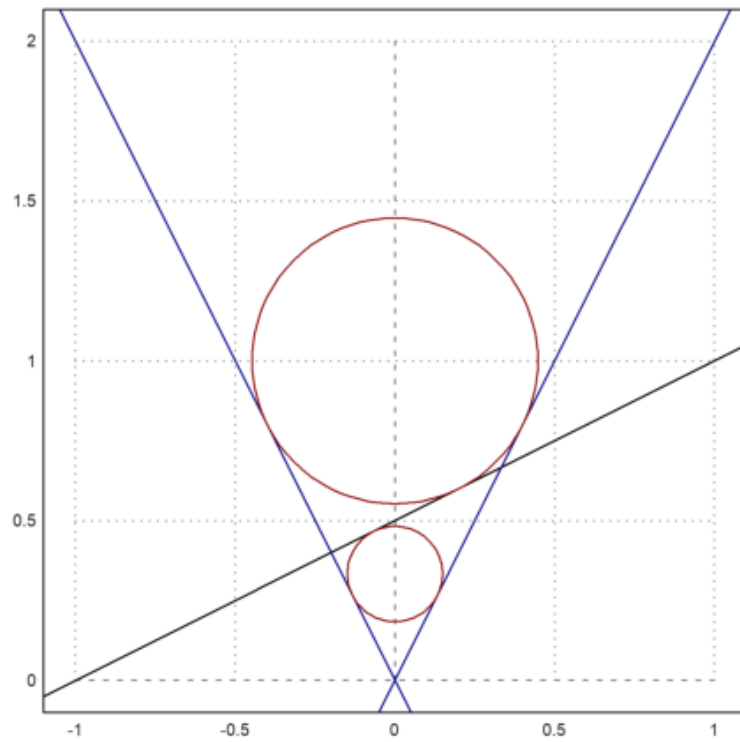
```
[0.149071, 0.447214]
```

Plot lingkaran ke dalam gambar

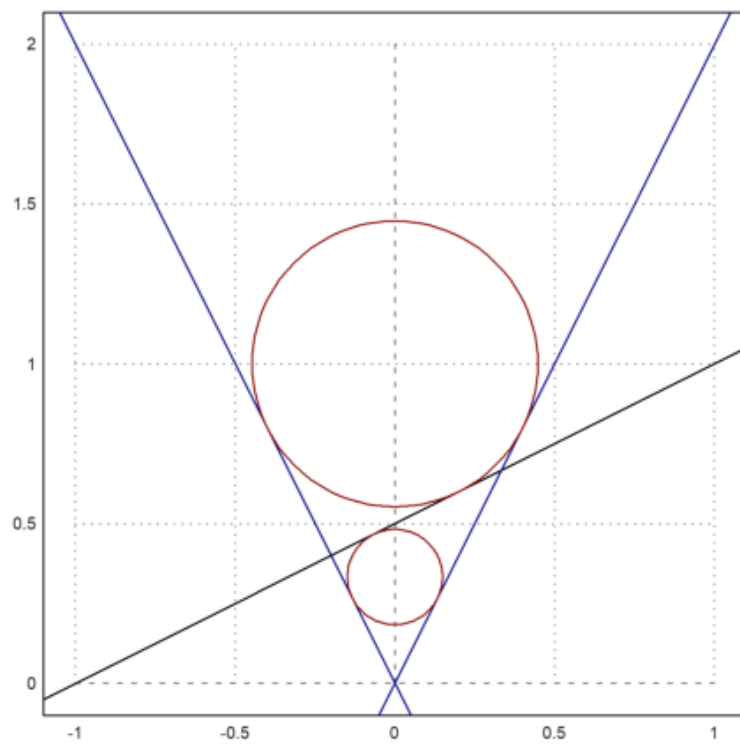
```
>color(red);  
>plotCircle(circleWithCenter([0,u[1]],dd[1]),""):
```



```
>plotCircle(circleWithCenter([0,u[2]],dd[2]),""):
```



```
>insimg;
```



Plot dengan Povray

Selanjutnya kami merencanakan semuanya dengan Povray. Perhatikan bahwa Anda mengubah perintah apa pun dalam urutan perintah Povray berikut, dan menjalankan kembali semua perintah dengan Shift-Return. Pertama kita memuat fungsi povray.

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

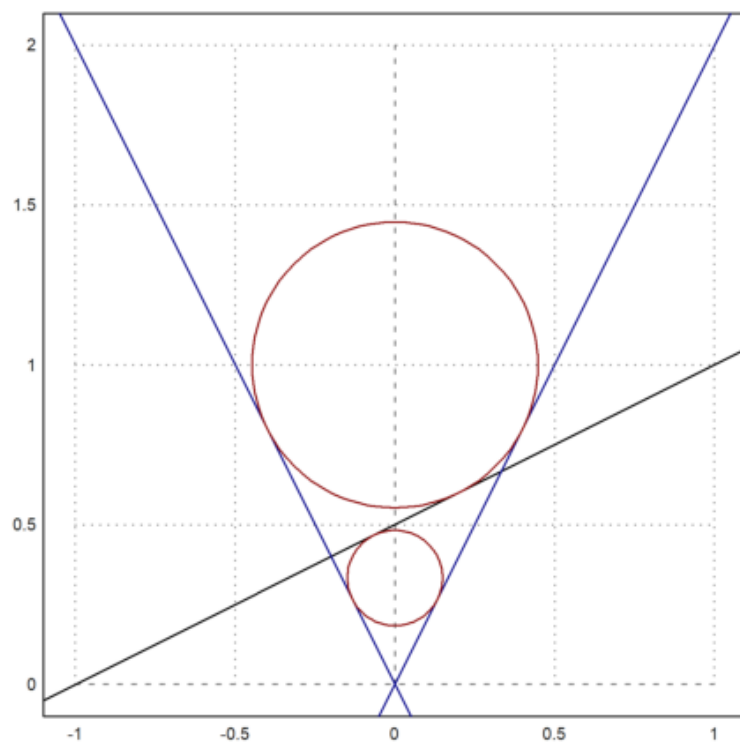
C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe

Kami mengatur adengan dengan tepat

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

Selanjutnya kita menulis dua bidang 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 transparan



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

Kami menghasilkan 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 pada lingkaran, dimana 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 kami menghasilkan dua titik dimana bola menyentuh bidang. Ini adalah fookus 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 menghitung perpotongan P1P2 dengan bidang

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

Kami menghubungkan titik-titik dengan segmen garis

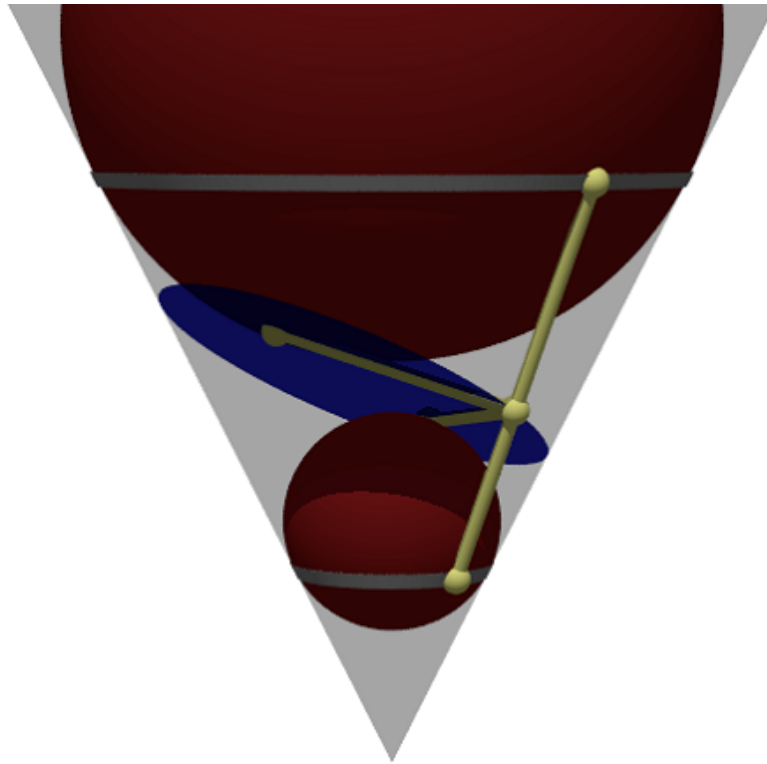
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan pira abu-abu dimana 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)));
```

## Mulai program Povray

```
>povend();
```



Untuk mendapatkan Anaglyph ini kita perlu memasukkan semuanya ke dalam fungsi scene. Fungsi ini akan digunakan dua kali kemudian.

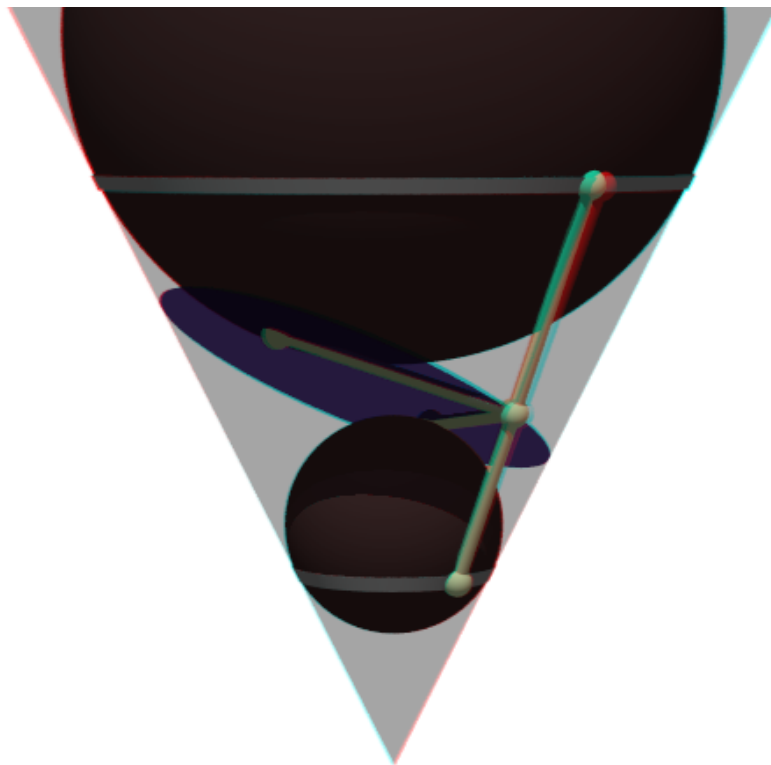
```
>function scene () ...
```

```
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)));
```

```
writeln(povsegment(P5,P3,povlook(yellow)));
writeln(povsegment(P5,P4,povlook(yellow)));
pcw=povcone([0,0,0],0,[0,0,a],1.01);
pcl=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pcl],povlook(gray)));
pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pc2],povlook(gray)));
endfunction
```

Anda membutuhkan kacamata merah/sian untuk menghargai efek berikut.

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



## Contoh 8: Geometri Bumi

Dalam buku catatan ini, kami ingin melakukan beberapa perhitungan sferis. Fungsi-fungsi tersebut terdapat dalam file "spherical.e" di folder contoh. Kita perlu memuat file itu terlebih dahulu.

```
>load "spherical.e";
```

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

```
>FMIPA=[rad(-7,-46.467),rad(110,23.05)]
```

```
[-0.13569, 1.92657]
```

Anda dapat mencetak posisi ini dengan sposprint (cetak posisi spherical).

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

```
S 7°46.467' E 110°23.050'
```

Mari kita 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)]
```

```
[-0.121897, 1.927]
```

```
>sposprint(Solo), sposprint(Semarang),
```

```
S 7°34.333' E 110°49.683'
```

```
S 6°59.050' E 110°24.533'
```

Pertama kita menghitung vektor dari satu ke yang lain pada bola ideal. Vektor ini [pos,jarak] dalam radian. Untuk menghitung jarak di bumi, kita kalikan dengan jari-jari bumi pada garis lintang 7°.

```
>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 berikut menggunakan perkiraan yang lebih baik. Pada jarak yang begitu pendek hasilnya hampir sama.

```
>esdist(FMIPA,Semarang)->" km" // perkiraan jarak FMIPA-Semarang
```

```
Commands must be separated by semicolon or comma!
```

```
Found: // perkiraan jarak FMIPA-Semarang (character 32)
```

```
You can disable this in the Options menu.
```

```
Error in:
```

```
esdist(FMIPA,Semarang)->" km" // perkiraan jarak FMIPA-Semarang ...
```

Ada fungsi untuk heading, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang canggih.

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

```
65.34°
```

Sudut segitiga melebihi 180° 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 kecil, ini tidak akurat karena kesalahan pengurangan dalam asum-pi.

```
>(asum-pi)*rearth(48°)^2->" km^2", // perkiraan luas segitiga FMIPA-Solo-Semarang
```

```
2116.02948749 km^2
```

Ada fungsi untuk ini, yang menggunakan garis lintang rata-rata segitiga untuk menghitung jari-jari bumi, dan menangani 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 dapat menambahkan vektor ke posisi. Sebuah vektor berisi heading dan jarak, keduanya dalam radian. Untuk mendapatkan vektor, kami menggunakan `vector`. Untuk menambahkan vektor ke posisi, kami menggunakan `sadd`.

```
>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 yang ideal. Hal yang sama 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 pendekatan yang baik.

```
>esdist(Tugu,Monas)->" km", // perkiraan jarak Tugu Jogja - Monas Jakarta
```

```
431.565659488 km
```

Judulnya sama dengan judul 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 semula. Hal ini terjadi, karena kita tidak menghitung fungsi invers secara tepat, tetapi mengambil perkiraan jari-jari bumi di sepanjang jalan.

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

```
S 6°10.500' E 106°48.717'
```

Namun, kesalahannya tidak besar

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

```
S 6°10.500' E 106°48.717'
```

Tentu kita tidak bisa berlayar dengan tujuan yang sama dari satu tujuan ke tujuan lainnya, jika kita ingin menempuh jalur terpendek. Bayangkan, Anda terbang NE mulai dari titik mana pun di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti heading yang konstan!

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

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

Sekarang kita tambahkan 10 kali sepersepuluh dari jarak, menggunakan pos ke Monas, kita sampai di Tugu.

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

Hasilnya jauh

```
>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 garis lintang yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

Jalur terpendek dari P1 ke P2 bukanlah lingkaran garis lintang 30°, melainkan jalur terpendek yang dimulai 10° lebih jauh ke utara di P1.

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

79.69°

Tapi, jika kita mengikuti pembacaan kompas ini, kita akan berputar ke kutub utara! Jadi kita harus menyesuaikan arah kita di sepanjang jalan. Untuk tujuan kasar, kami menyesuaikannya pada 1/10 dari total jarak.

```
>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 tepat, karena kita akan menambahkan sedikit kesalahan, jika kita mengikuti heading yang sama terlalu lama.

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

0.203km

Kami mendapatkan perkiraan yang baik, jika kami menyesuaikan pos setelah setiap 1/100 dari total jarak dari Tugu ke 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 keperluan navigasi, kita bisa mendapatkan urutan posisi GPS di sepanjang lingkaran besar menuju Monas dengan fungsi navigasi.

```
>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'

Kami menulis sebuah fungsi, yang memplot bumi, dua posisi, dan posisi di antaranya.

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

```
useglobal;  

plotearth;  

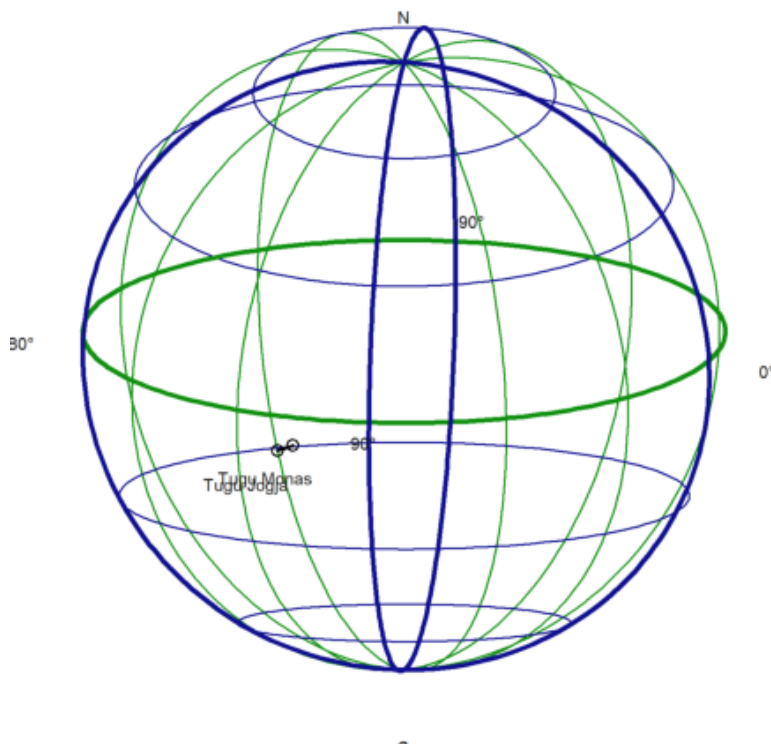
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");  

plotposline(v);  

endfunction
```

Sekarang kita plot semuanya

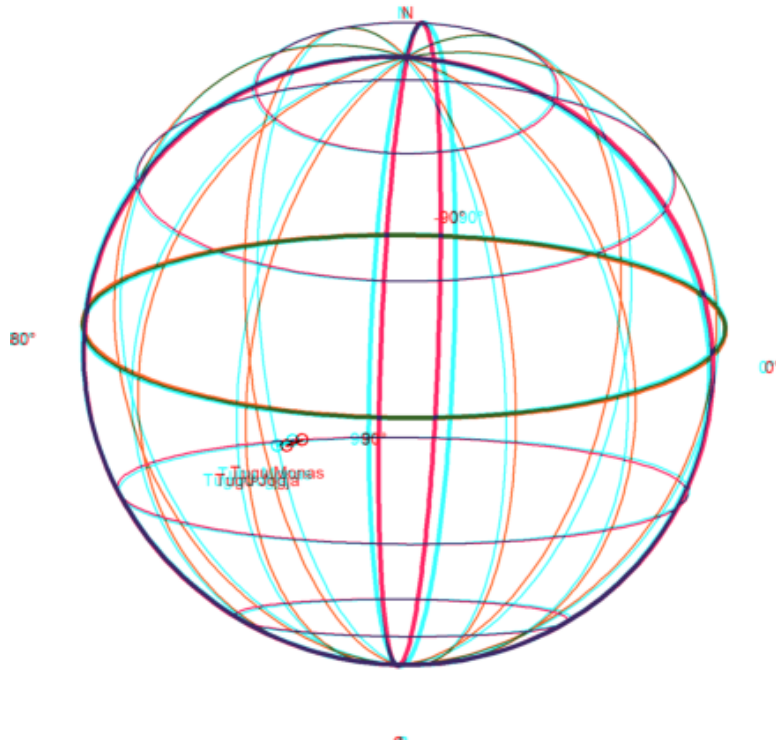
```
>plot3d("testplot",angle=25, height=6,>own,>user, zoom=4) :
```



Atau gunakan plot3d untuk mendapatkan tampilan anaglyph. Ini terlihat lebih baik 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),  $r$ .

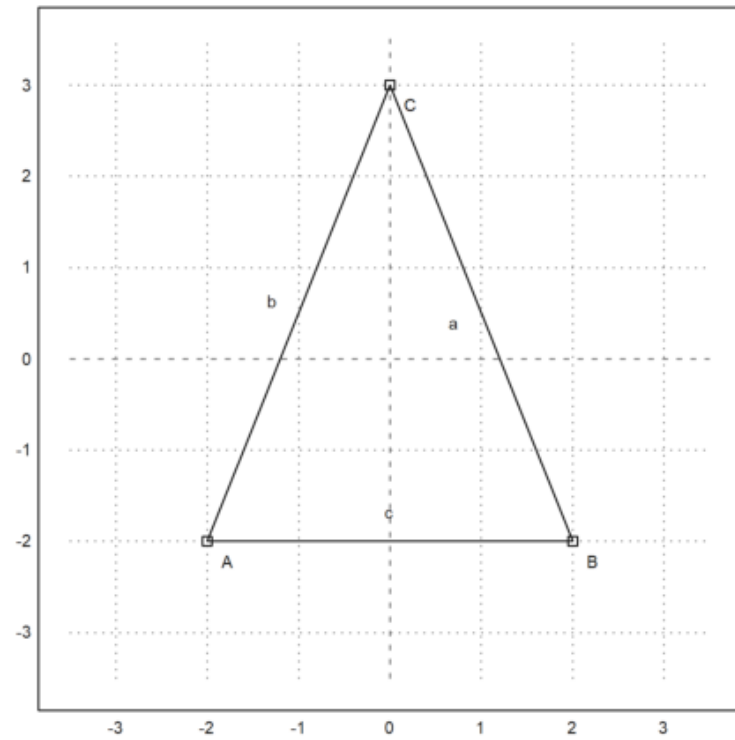
Petunjuk:

- 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.

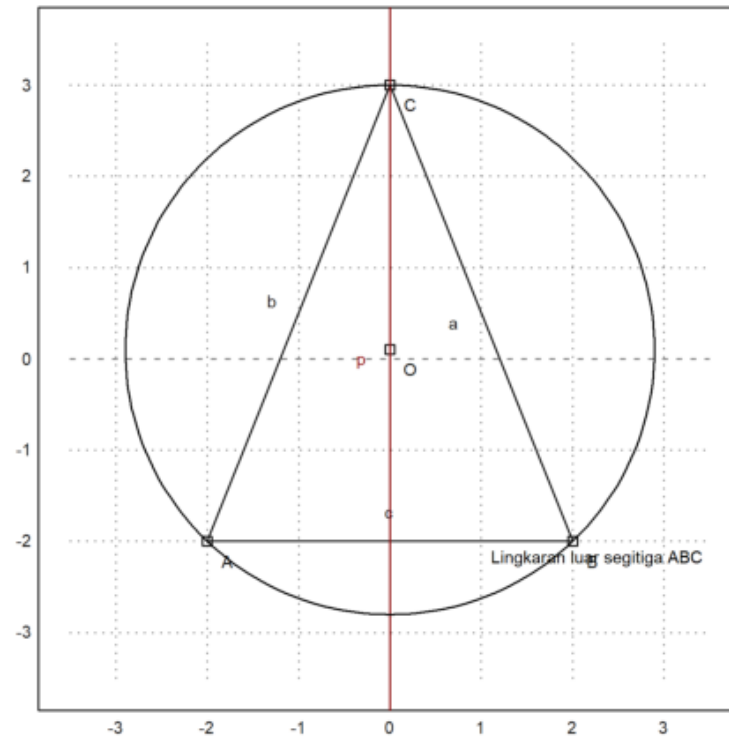
```
>load geometry
```

Numerical and symbolic geometry.

```
>setPlotRange(-3.5,3.5,-3.5,3.5);
>A=[-2,-2]; plotPoint(A,"A");
>B=[2,-2]; plotPoint(B,"B");
>C=[0,3]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b");
>aspect(1):
```



```
>c=circleThrough(A,B,C);
>R=getCircleRadius(c);
>O=getCircleCenter(c);
>plotPoint(O,"O");
>l=angleBisector(A,C,B);
>color(2); plotLine(l); color(1);
>plotCircle(c,"Lingkaran luar segitiga ABC");
```

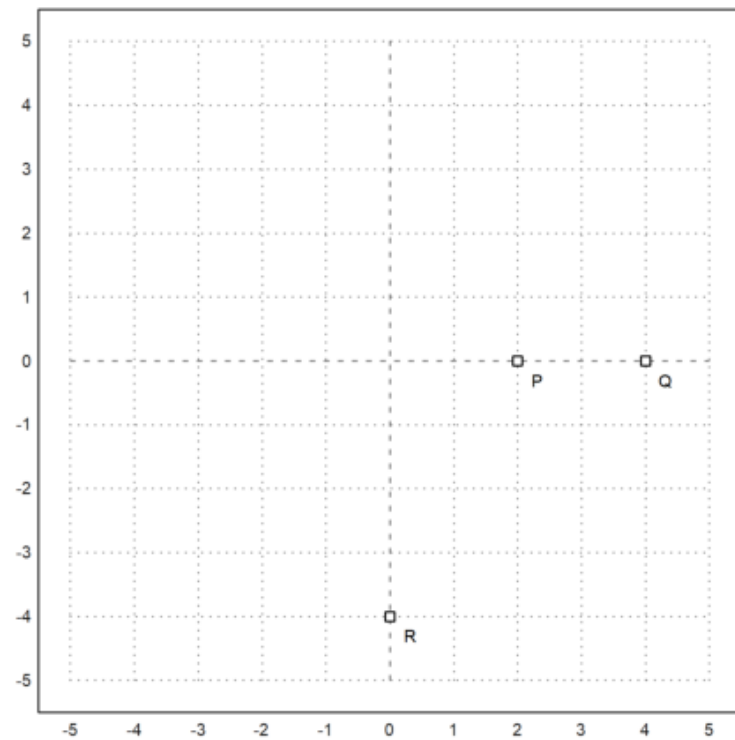


2. Gambarlah suatu parabola yang melalui 3 titik yang diketahui.

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  $a, b, c$ .

```
>load geometry;
>setPlotRange(5); P=[2,0]; Q=[4,0]; R=[0,-4];
>plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R"):
```



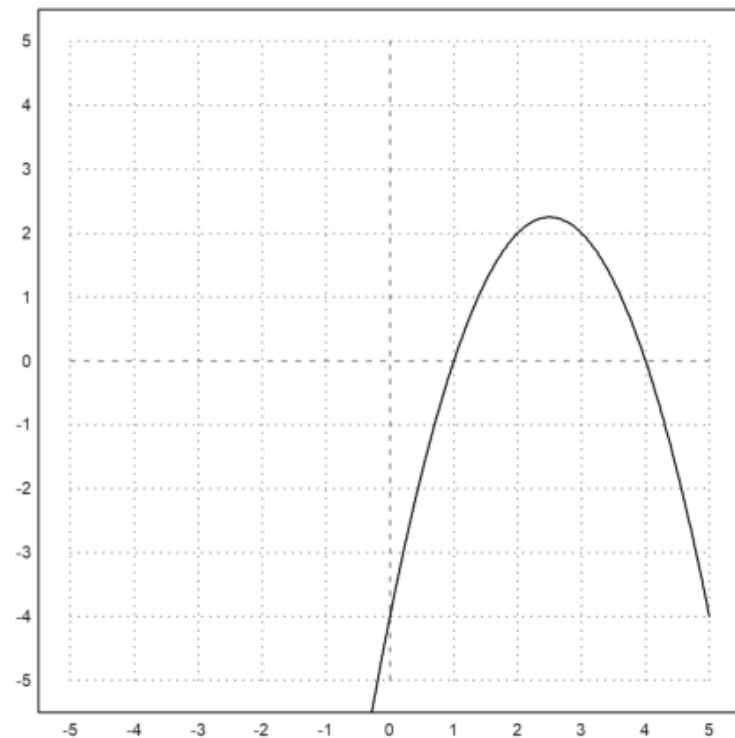
```
>sol &= solve([a+b=-c,16*a+4*b=-c,c=-4],[a,b,c])
```

```
[[a = - 1, b = 5, c = - 4]]
```

```
>function y&=-x^2+5*x-4
```

$$-x^2 + 5x - 4$$

```
>plot2d("-x^2+5*x-4",-5,5,-5,5):
```



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 kali panjang sisi-sisi yang berhadapan sama.

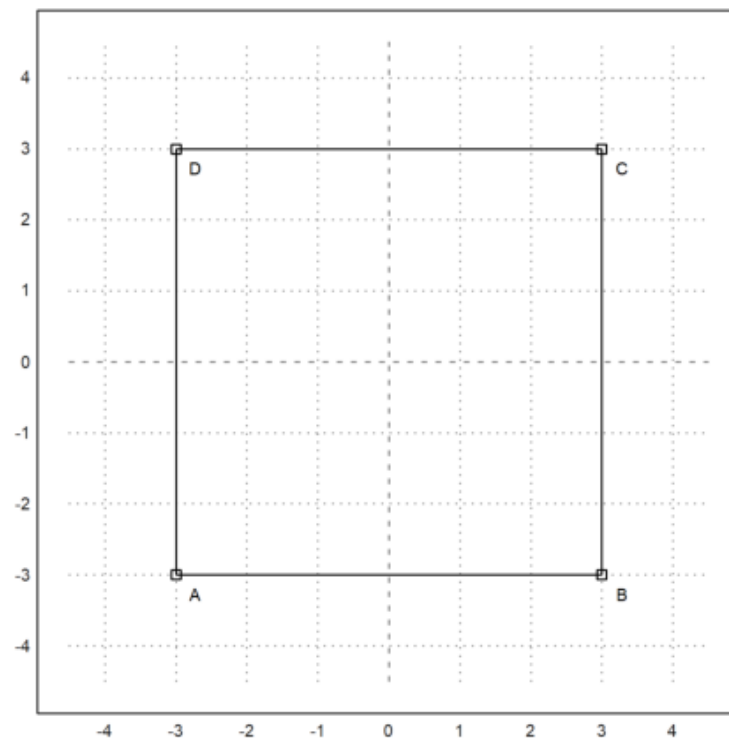
```
>load geometry
```

Numerical and symbolic geometry.

```

>setPlotRange(-4.5,4.5,-4.5,4.5);
>A=[-3,-3]; plotPoint(A,"A");
>B=[3,-3]; plotPoint(B,"B");
>C=[3,3]; plotPoint(C,"C");
>D=[-3,3]; plotPoint(D,"D");
>plotSegment(A,B,"");
>plotSegment(B,C,"");
>plotSegment(C,D,"");
>plotSegment(A,D,"");
>aspect(1):

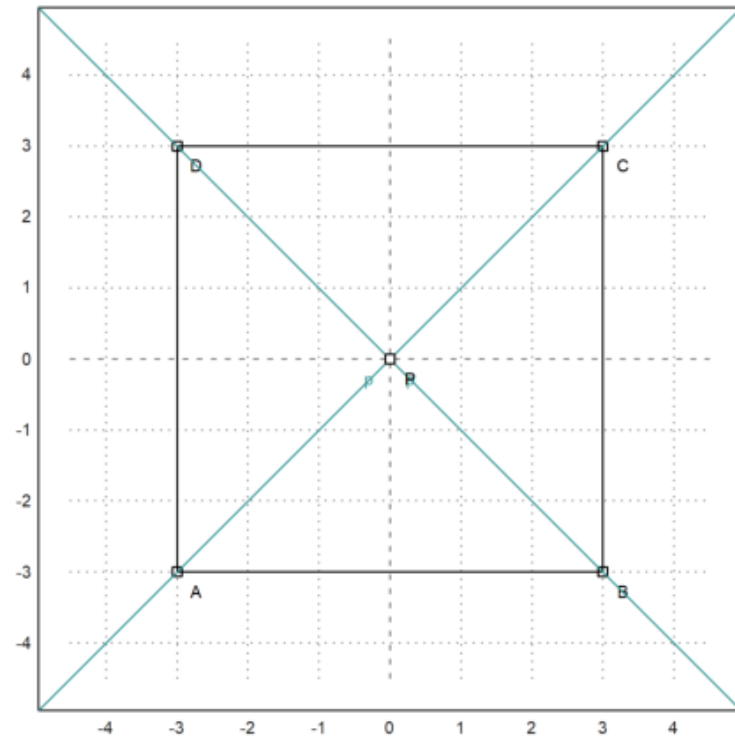
```



```

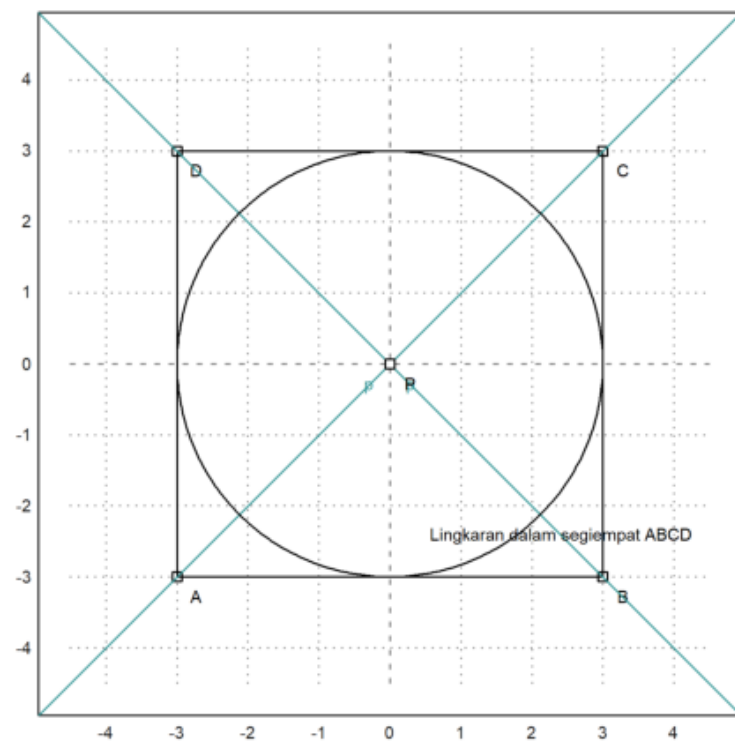
>l=angleBisector(A,B,C);
>m=angleBisector(B,C,D);
>P=lineIntersection(l,m);
>color(5); plotLine(l); plotLine(m); color(1);
>plotPoint(P,"P"):

```



Dari gambar diatas terlihat bahwa keempat garis bagi sudutnya bertemu di satu titik yaitu titik P.

```
>r=norm(P-projectToLine(P,lineThrough(A,B)));
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segiempat ABCD"):
```



Dari gambar diatas, terlihat bahwa sisi-sisinya merupakan garis singgung lingkaran yang sama yaitu lingkaran dalam segiempat.

Akan ditunjukkan bahwa hasil kali panjang sisi-sisi yang berhadapan sama.

```
>AB=norm(A-B) //panjang sisi AB
```

6

```
>CD=norm(C-D) //panjang sisi CD
```

6

```
>AD=norm(A-D) //panjang sisi AD
```

6

```
>BC=norm(B-C) //panjang sisi BC
```

6

```
>AB.CD
```

36

```
>AD.BC
```

36

Terbukti bahwa hasil kali panjang sisi-sisi yang berhadapan sama yaitu 36. Jadi dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

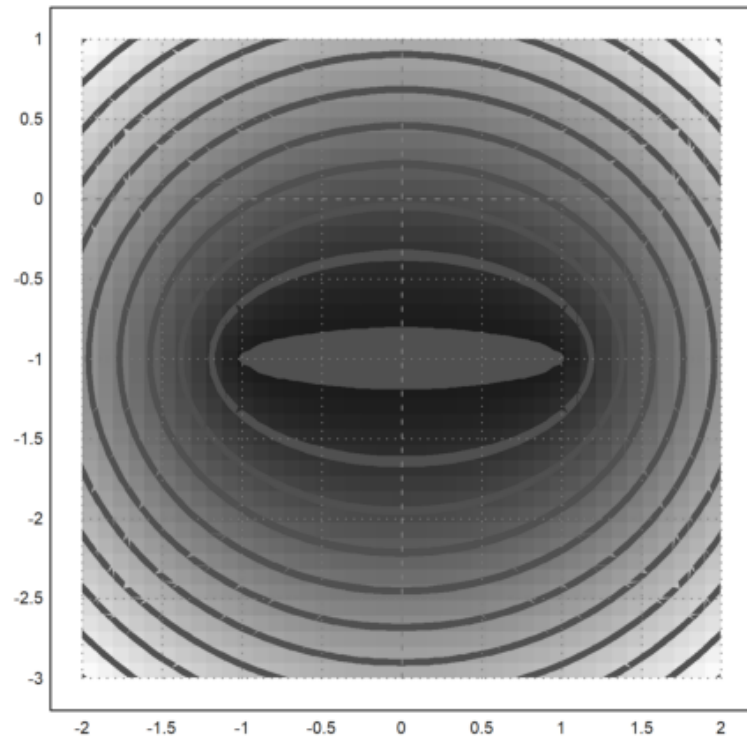
4. Gambarkanlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).

Penyelesaian :

Diketahui kedua titik fokus P = [-1,-1] dan Q = [1,-1]

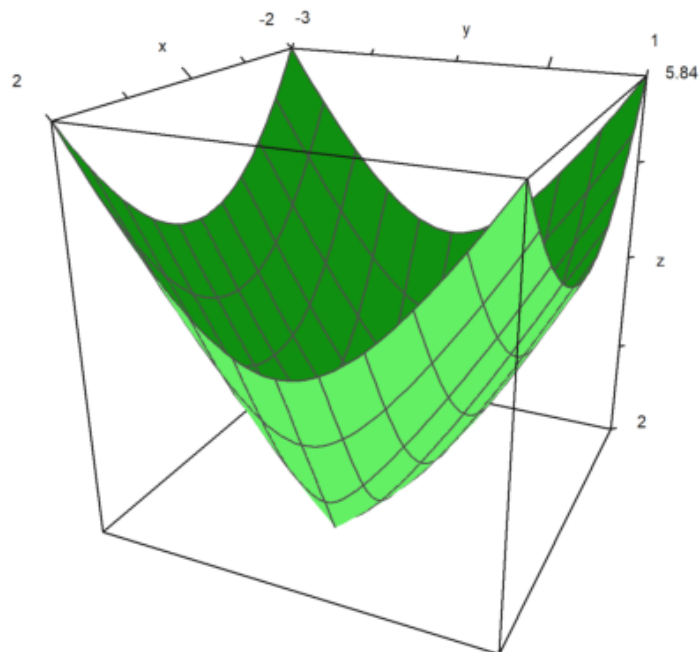
```
>P=[-1,-1]; Q=[1,-1];  
>function d1(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)  
>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)+sqrt((x-Q[1])^2+(y-Q[2])^2)  
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```





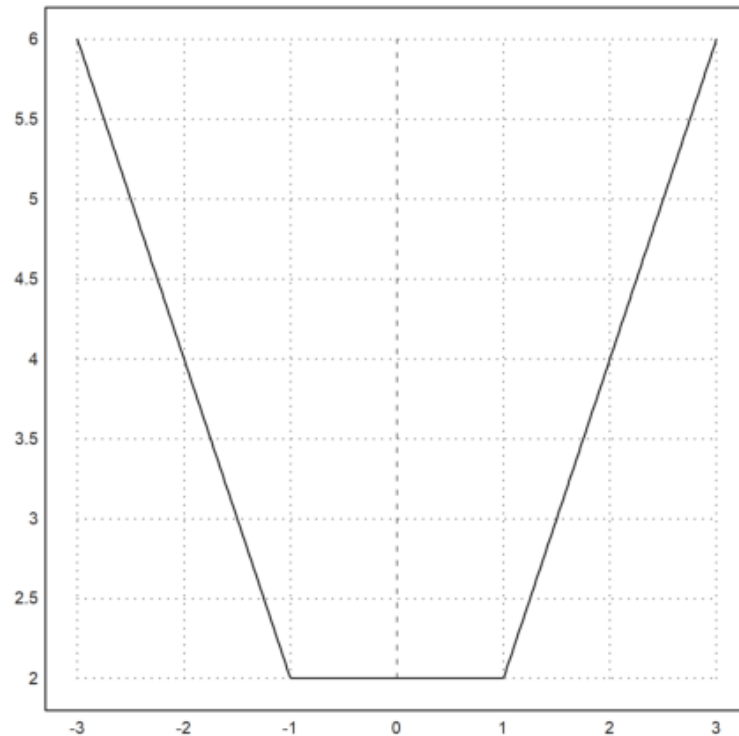
Grafik yang lebih menarik

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



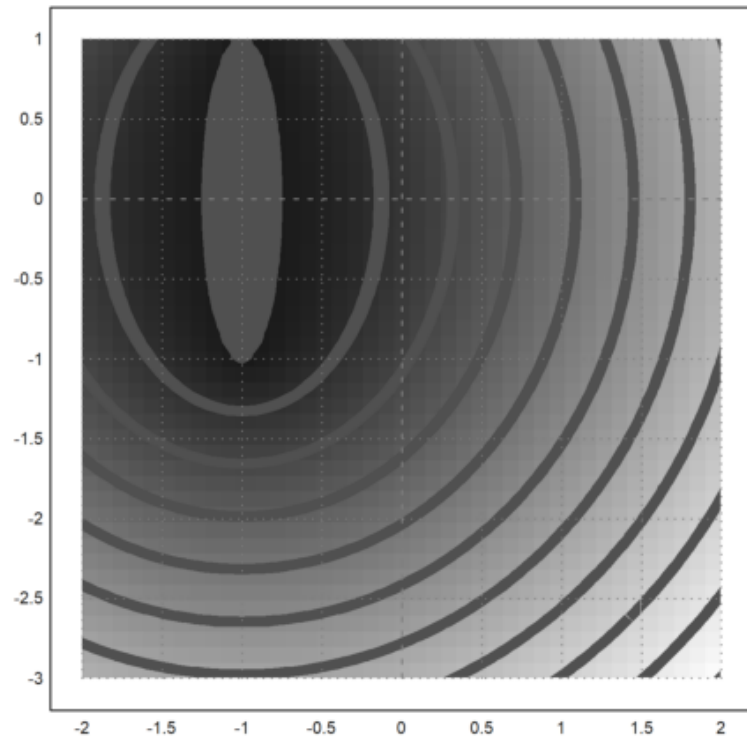
## Batasan ke garis PQ

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

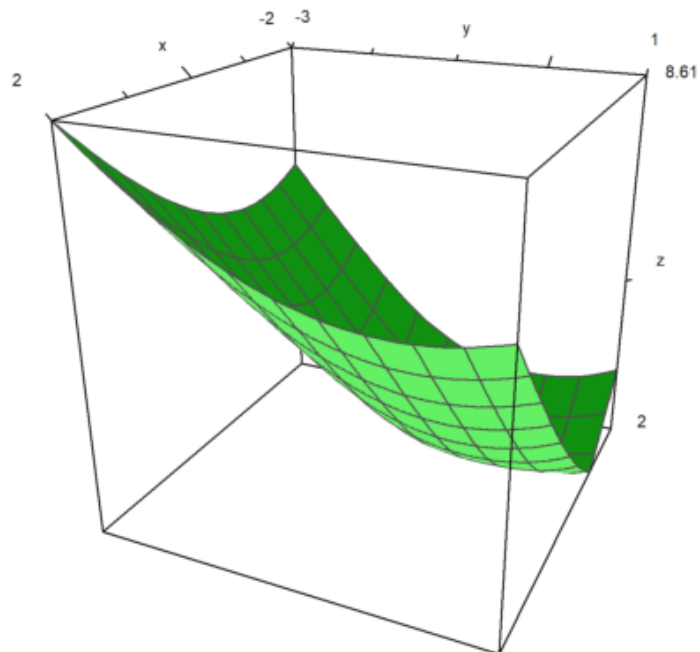


5. Gambarlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

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



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



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

