

Nama: Lathif Nur Rahman
Kelas: Matematika E
NIM: 23030630077

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 <ABC
angleBisector(A, B, C): garis bagi sudut <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  $sa=\sin(\phi)^2$  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 tetapkan tiga titik dan gambarkan.

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

Lalu tiga segmen.

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

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

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

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

Hitung garis tegak lurus yang melalui A di BC.

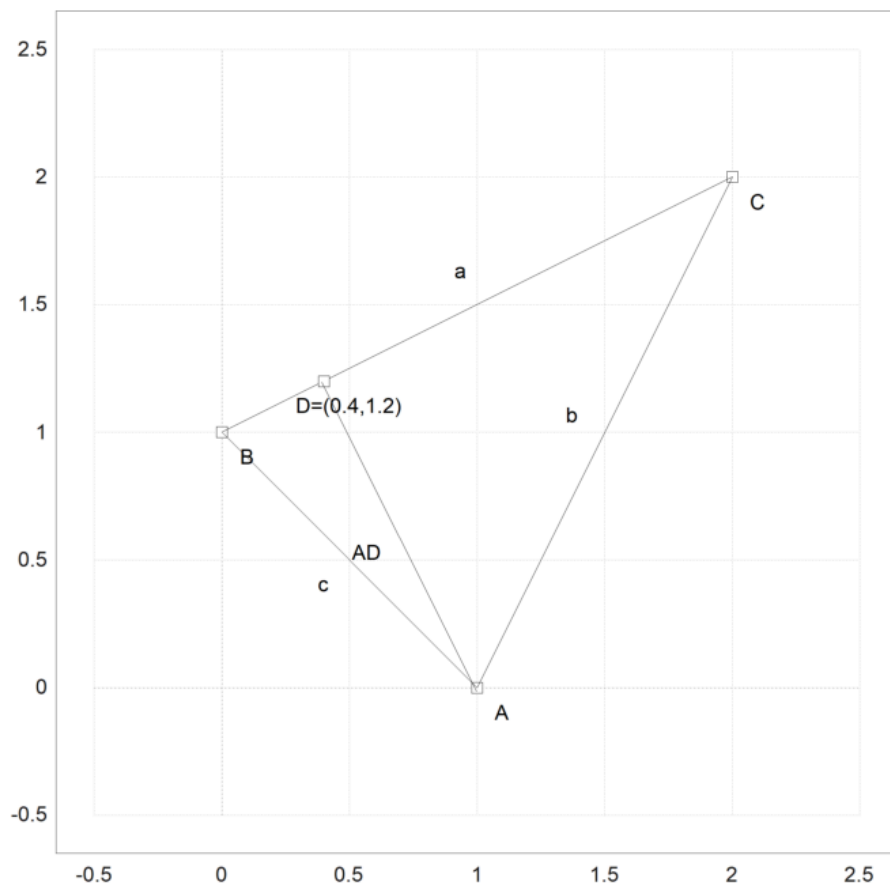
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan persimpangannya dengan SM.

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

Gambarkan itu

```
>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)
```

Bandungkan 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

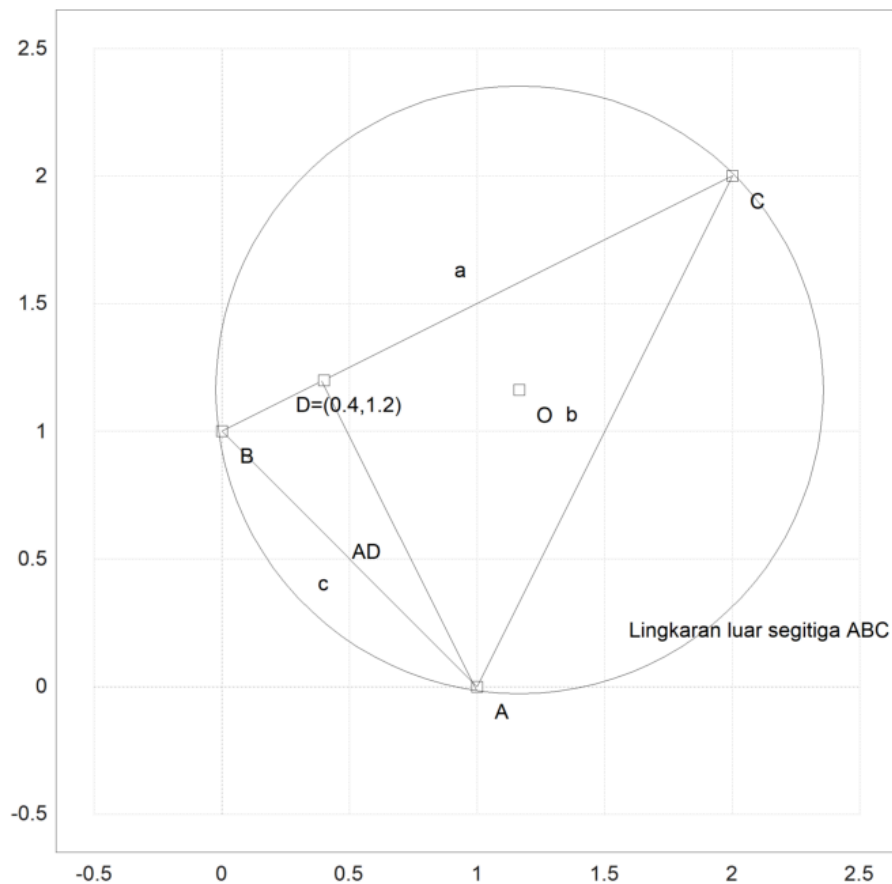
Sudut di C.

```
>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
```

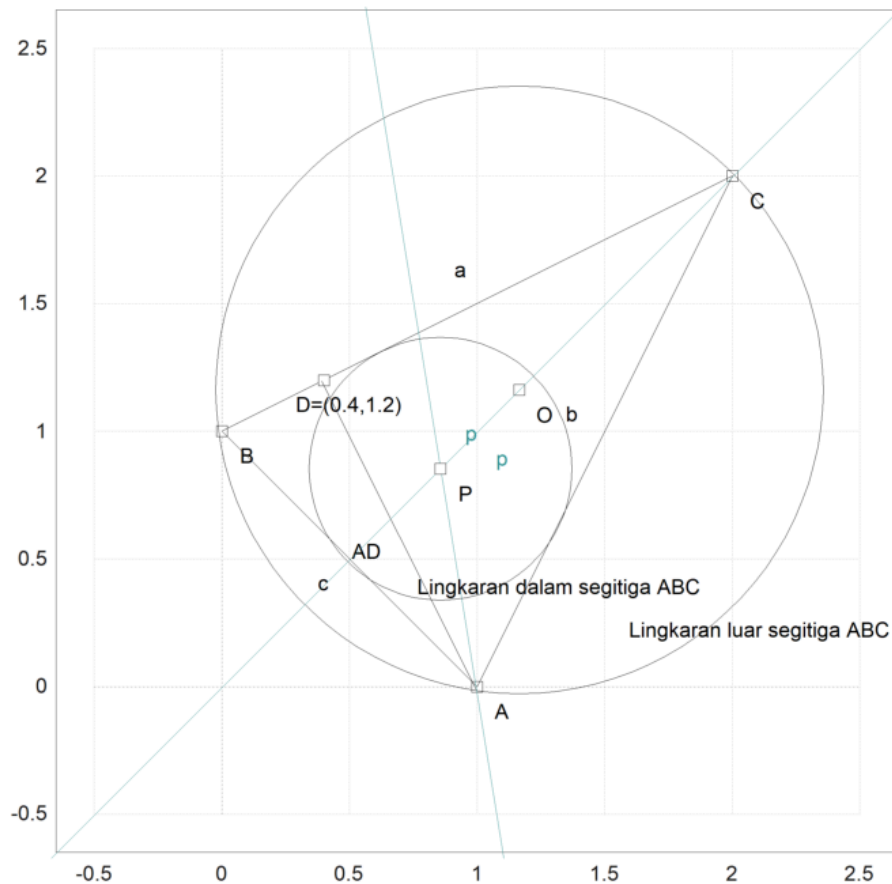
```
[0.86038, 0.86038]
```

Tambahkan 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 dalam
```



Latihan

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

```
>d=lineCircleIntersections(lineThrough(C,B), circleWithCenter(P,r))
```

```
[0.632456, 1.31623]
```

```
>e=lineCircleIntersections(lineThrough(C,A), circleWithCenter(P,r))
```

```
[1.31623, 0.632456]
```

```
>f=lineCircleIntersections(lineThrough(A,B), circleWithCenter(P,r))
```

```
[0.5, 0.5]
```

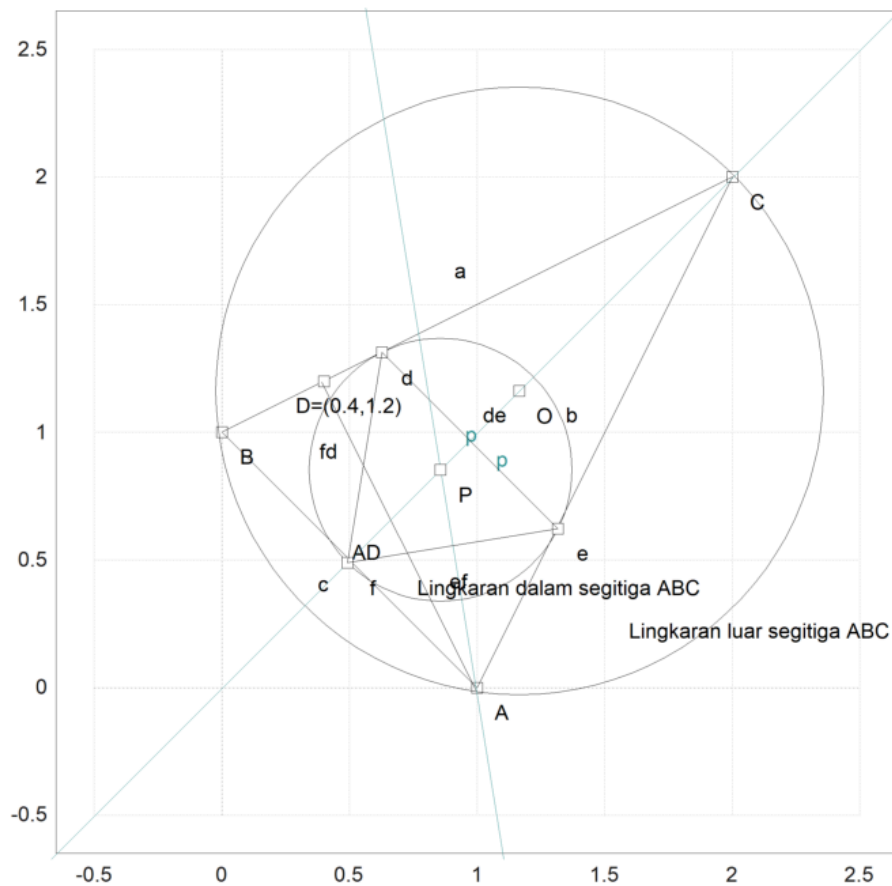
2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?

```
>plotPoint(d);  
>plotPoint(e);
```

```

>plotPoint(f);
>plotSegment (d,e, "de");
>plotSegment (f,d, "fd");
>plotSegment (e,f, "ef");

```



```

>distance(d, e)

```

0.966999966873

```

>distance(e, f)

```

0.826905214631

```

>distance(f, d)

```

0.826905214631

Membentuk Segitiga sama kaki

3. Hitung luas segitiga tersebut.

```

>areaTriangle(d, e, f)

```

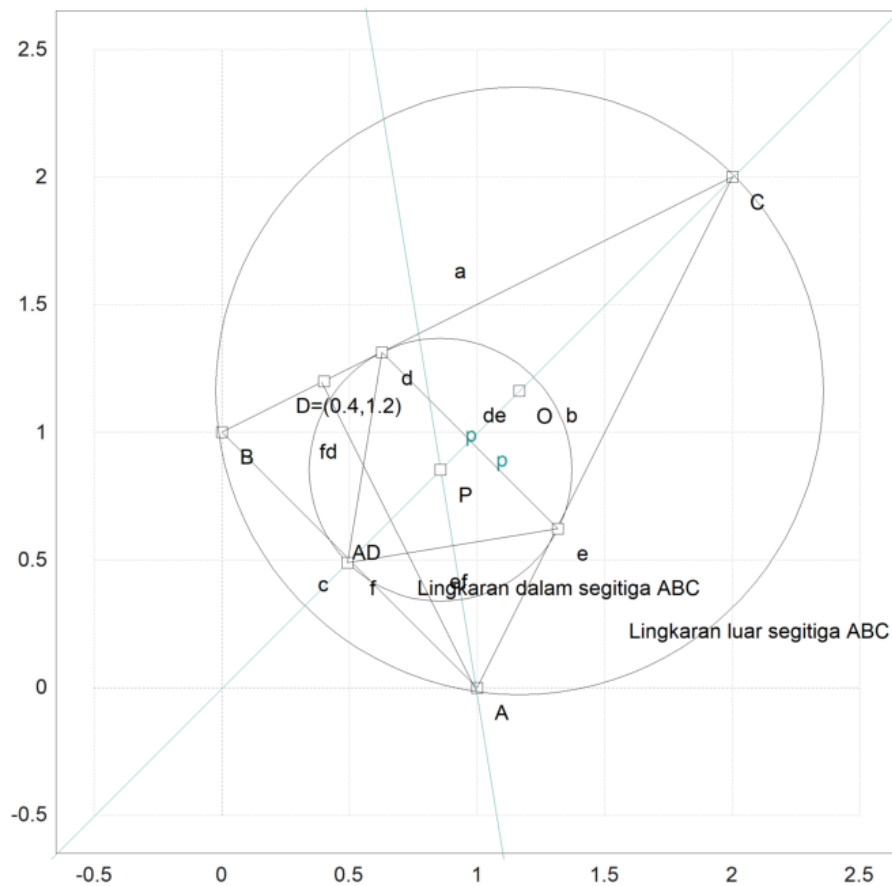
0.324341649025

Manual

```

>i=perpendicular(f,lineThrough(d,e));
>j=lineIntersection(i,lineThrough(d,e));

```



```
>distance(f,j)
```

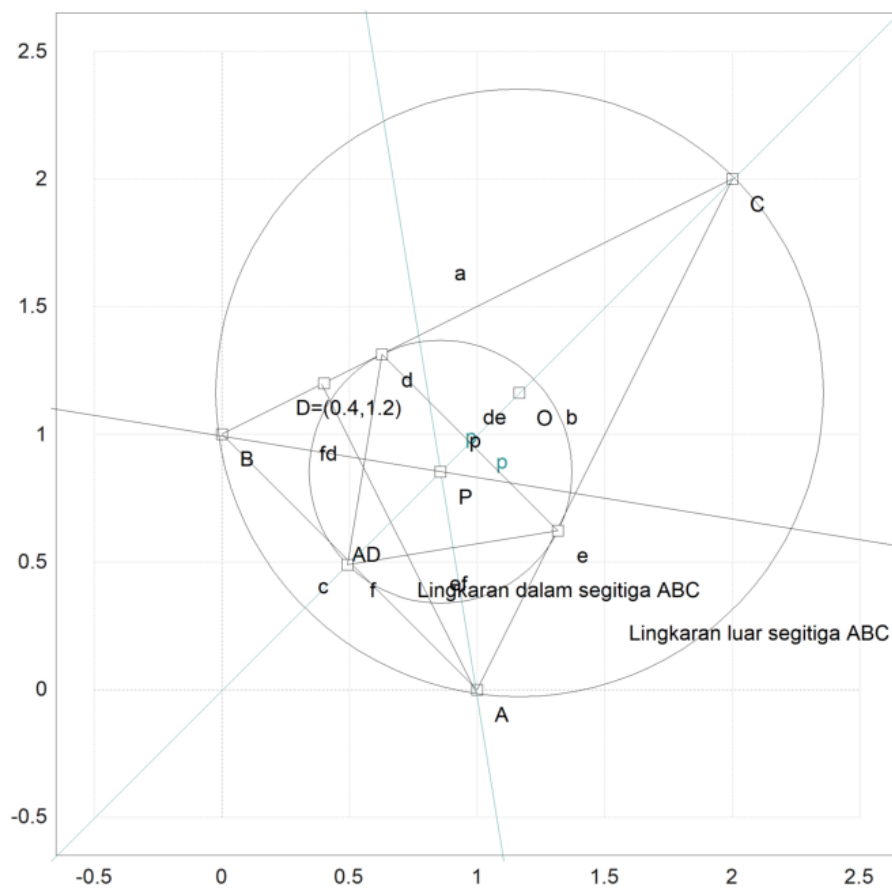
```
0.67082039325
```

```
>distance (f,j)*distance(d, e)*0.5 //Luas Segitiga
```

```
0.324341649025
```

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

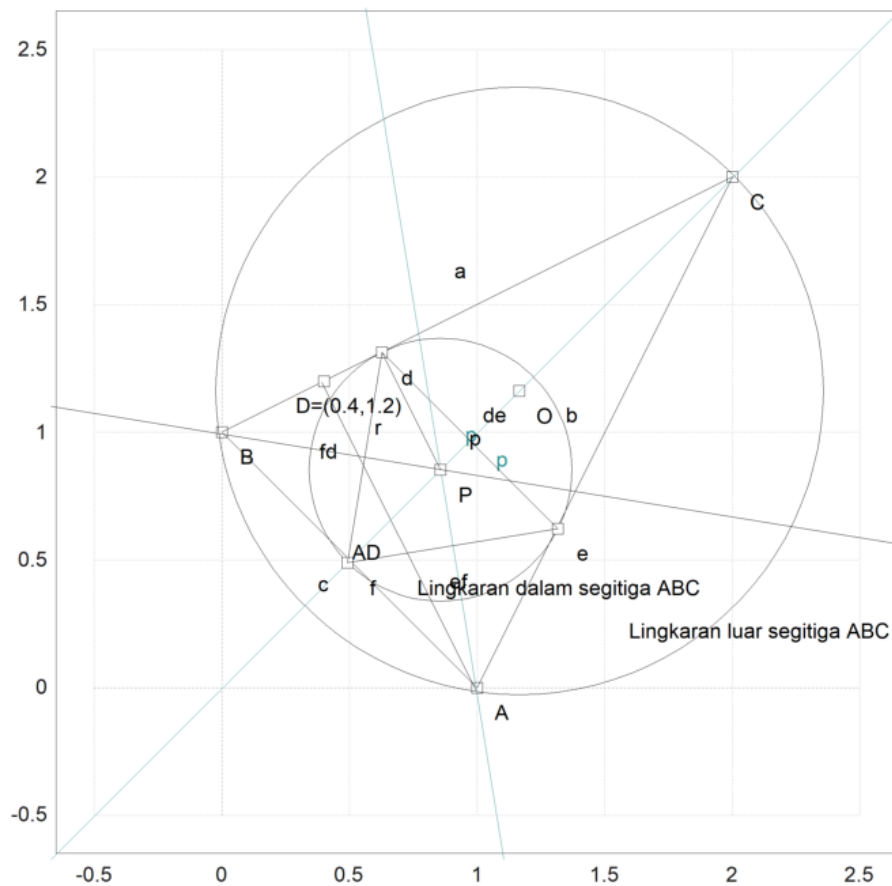
```
>k=angleBisector(C,B,A);  
>plotLine(k):
```



Terlihat dengan jelas

5. Gambar jari-jari lingkaran dalam.

```
>plotSegment (P,d, "r"):
```



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

```
>function f(r) &= pi*r^2; $&f(r)
```

$$\pi r^2$$

```
>f(R) //luas lingkaran luar
```

```
4.36332312999
```

```
>f(distance(P,d)) //luas lingkaran dalam
```

```
0.81601903655
```

Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

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

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

Fungsi garis dan lingkaran berfungsi sama seperti fungsi Euler, namun menyediakan komputasi simbolik.

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

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

Kita bisa mendapatkan persamaan garis dengan mudah.

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

$$2y - x = 2$$

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

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

$$x \left(y_1 - y_2 \right) + \left(x_2 - x_1 \right) y = x_1 \left(y_1 - y_2 \right) + \left(x_2 - x_1 \right) y_1$$

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

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

$$\left(x_1 - 1 \right) y - x y_1 = -y_1$$

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

$$\left[2, 1, 2 \right]$$

```
>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
```

$$\frac{5}{3\sqrt{2}}$$

$$1.178511301977579$$

```
>$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 garis bagi sudut
```

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

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

$$\left[0.86038, 0.86038 \right]$$

Perpotongan Garis dan Lingkaran

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

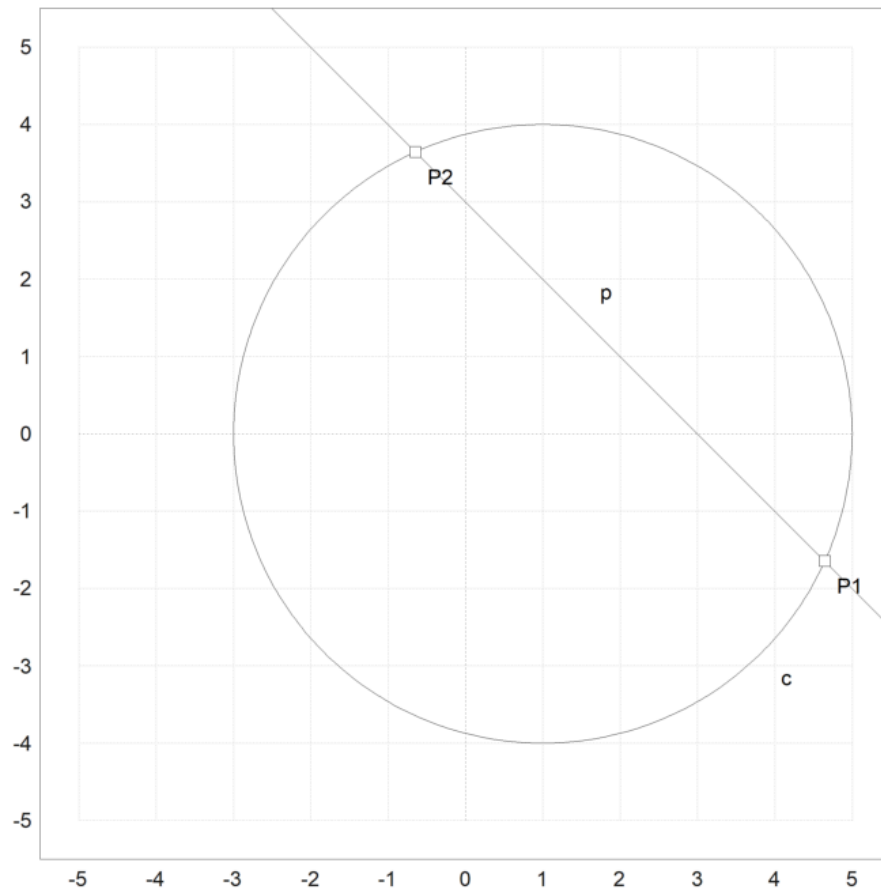
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

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

```
>{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 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
```

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

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

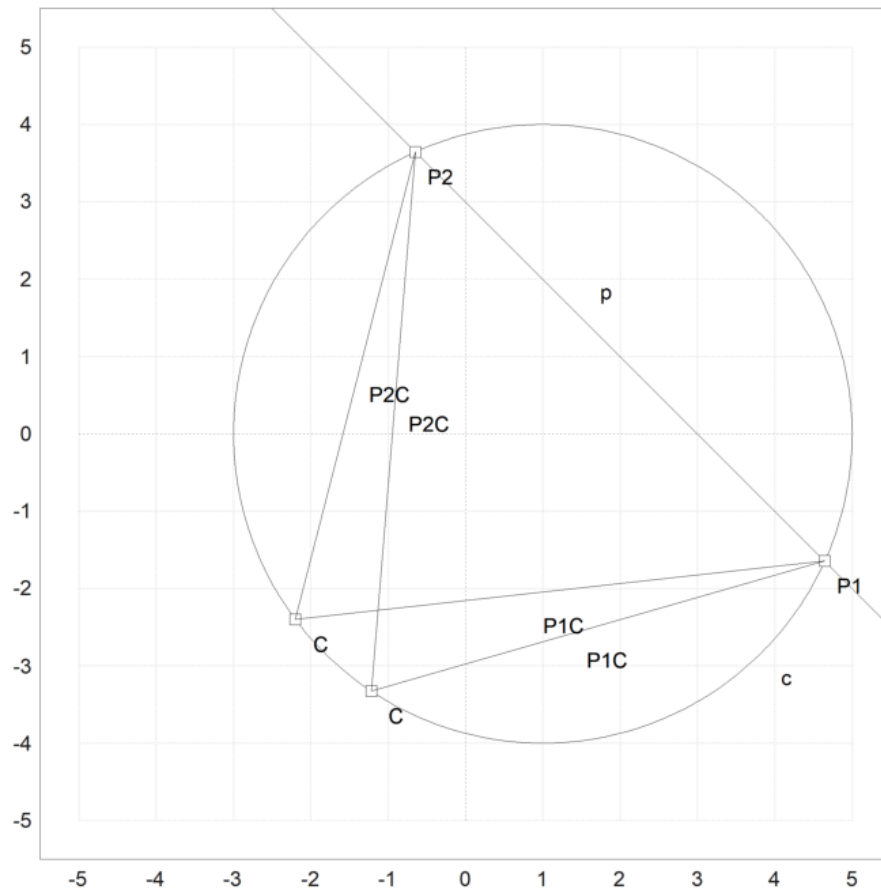
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
>degprint(computeAngle(P1,C,P2))
```

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;
```

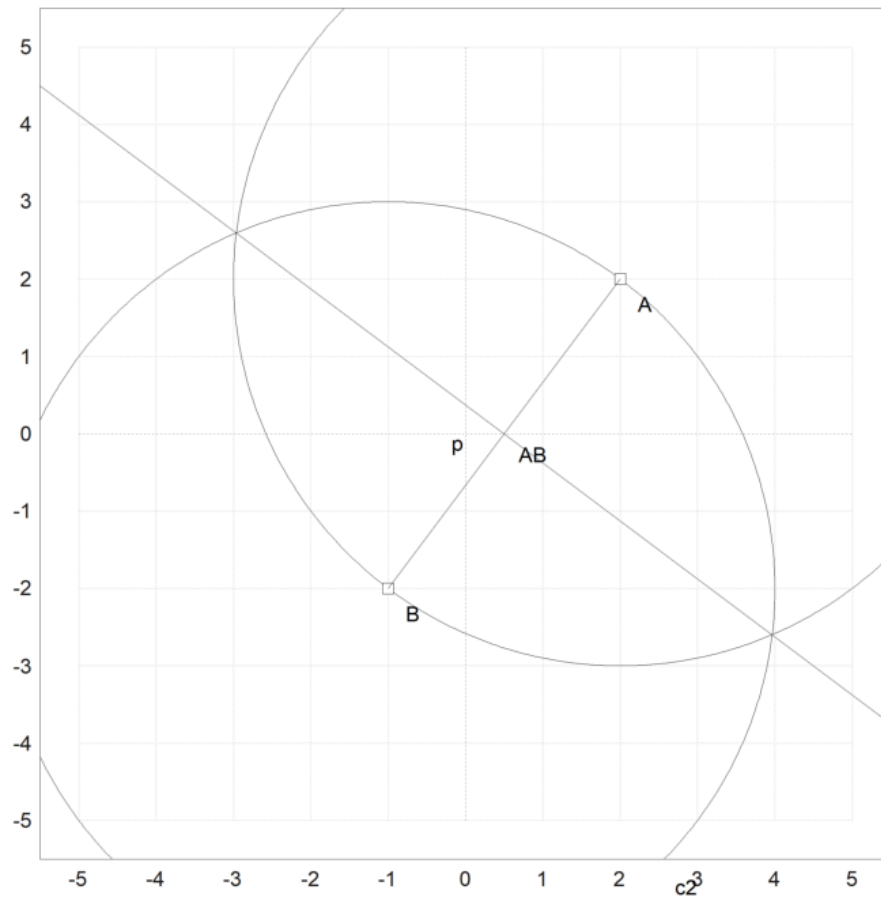


Garis Sumbu

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);
```



Selanjutnya kita melakukan 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 persimpangan cukup rumit. Tapi kita bisa menyederhanakannya jika kita mencari 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 tengah tegak lurus, yang dihitung dengan cara yang sangat 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:

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

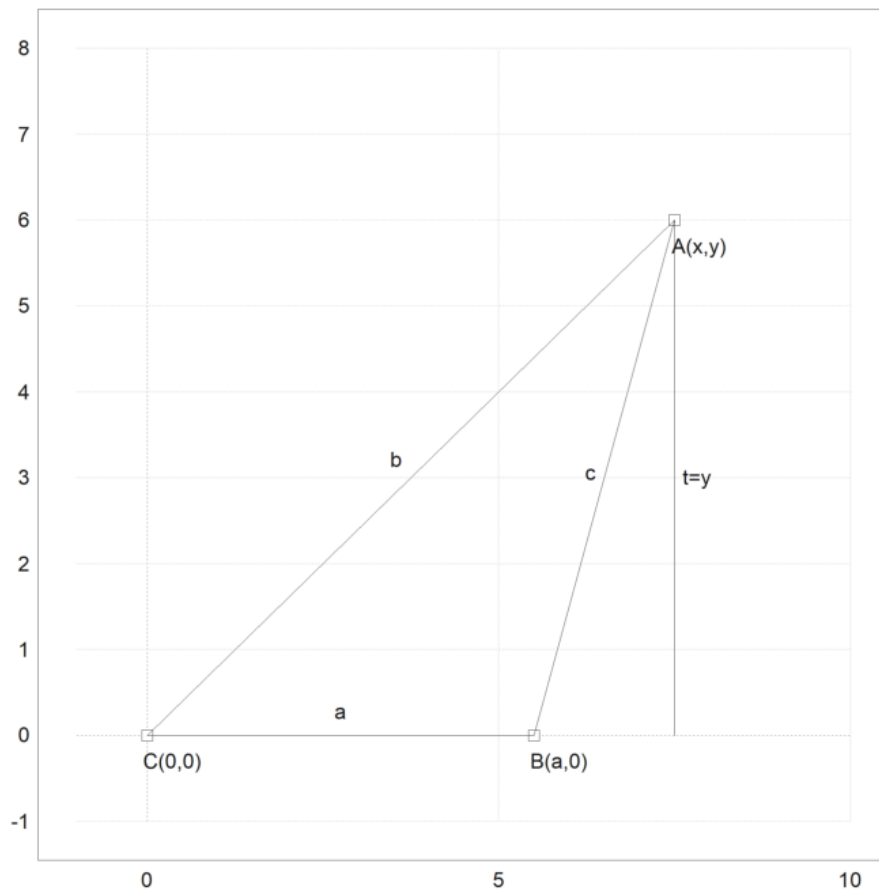
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

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])
```

[]

Ekstrak solusi y.

```
>ysol &= y with sol[2][2]
```

```
Maxima said:
part: invalid index of list or matrix.
-- an error. To debug this try: debugmode(true);
```

```
Error in:
ysol &= y with sol[2][2] ^
^
```

```
>$'y=sqrt(factor(ysol^2))
```

$$y = |ysol|$$

Kami mendapatkan rumus Heron.

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

$$H(a,b,[1,0,4]) = \frac{a |ysol|}{2}$$

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

$$Luas = |ysol|$$

Tentu saja, setiap segitiga siku-siku adalah kasus yang terkenal.

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

```
Variable or function ysol not found.  
Try "trace errors" to inspect local variables after errors.  
H:  
  useglobal; return a*abs(ysol)/2  
Error in:  
H(3,4,5) //luas segitiga siku-siku dengan panjang sisi 3, 4, 5 ...  
^
```

Dan jelas juga bahwa ini adalah segitiga dengan luas maksimal dan kedua sisinya 3 dan 4.

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

```
Variable or function ysol not found.  
Error in expression: 3*abs(ysol)/2  
%ploteval:  
  y0=f$(x[1],args());  
adaptiveevalone:  
  s=%ploteval(g$,t,args());  
Try "trace errors" to inspect local variables after errors.  
plot2d:  
  dw/n,dw/n^2,dw/n,auto,args();
```

Kasus umum juga berhasil.

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

```
Maxima said:  
diff: second argument must be a variable; found [1,0,4]  
-- an error. To debug this try: debugmode(true);  
  
Error in:  
$solve(diff(H(a,b,c)^2,c)=0,c) ...  
^
```

Sekarang mari kita cari himpunan semua titik di mana $b+c=d$ untuk suatu konstanta d . Diketahui bahwa ini adalah elips.

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

```
Maxima said:  
part: invalid index of list or matrix.  
-- an error. To debug this try: debugmode(true);  
  
Error in:  
s1 &= subst(d-c,b,sol[2]); $s1 ...  
^
```

Dan buatlah fungsinya.

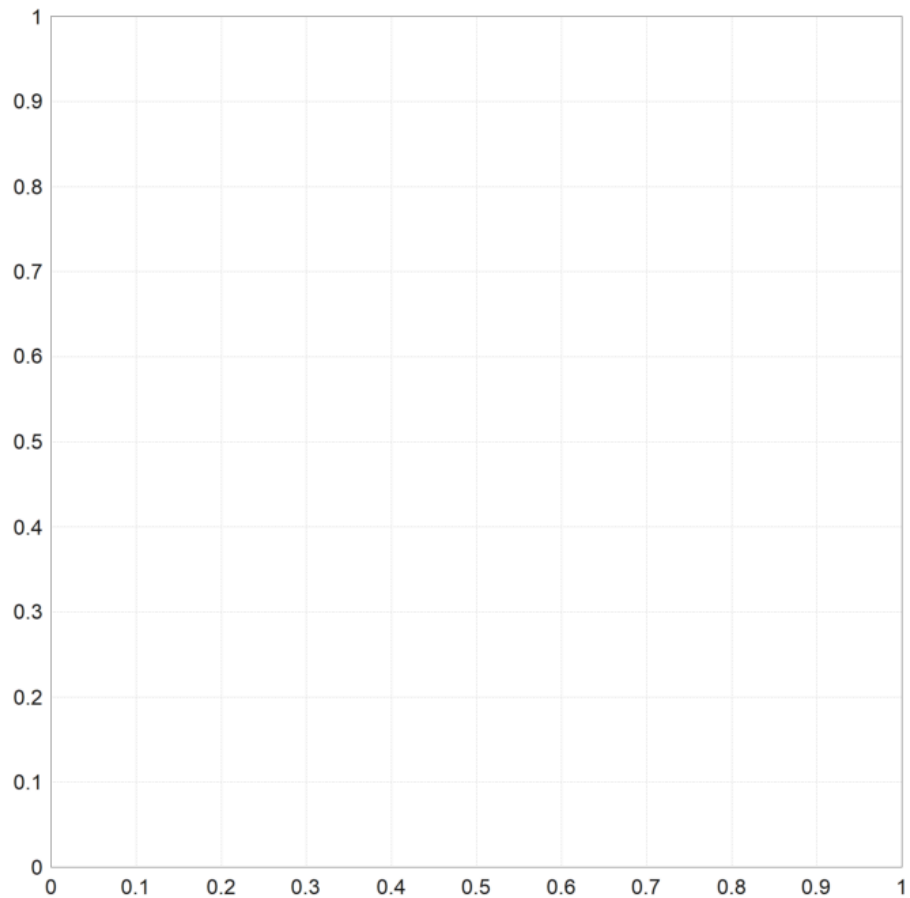
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

0

0

Sekarang kita bisa menggambar setnya. Sisi b bervariasi dari 1 sampai 4. Diketahui bahwa kita memperoleh elips.

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



Kita dapat memeriksa persamaan umum elips ini, 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 dan v adalah setengah 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])
```

$$\frac{a^2}{d^2}$$

Kita melihat bahwa tinggi dan luas segitiga adalah maksimal untuk $x=0$. Jadi luas segitiga dengan $a+b+c=d$ adalah maksimal jika segitiga tersebut sama sisi. Kami ingin memperolehnya secara analitis.

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

$$\left[\frac{a^2}{2} = 0, 0 = 0 \right]$$

Kita mendapatkan nilai minimum yang dimiliki oleh segitiga dengan salah satu sisinya 0, dan solusinya $a=b=c=d/3$.

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

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

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

```
>&solve([diff(H(a,b,c)^2,a)=1a,diff(H(a,b,c)^2,b)=1a, diff(H(a,b,c)^2,c)=1a,a+b+c=d],[a,b,c,1a])
```

```
Maxima said:
diff: second argument must be a variable; found [1,0,4]
-- an error. To debug this try: debugmode(true);

Error in:
... diff(H(a,b,c)^2,b)=1a, diff(H(a,b,c)^2,c)=1a,a+b+c=d],[a,b,c,1a]) ...
```

Kita bisa membuat plot situasinya

Pertama atur poin di Maxima.

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

```
Maxima said:  
part: invalid index of list or matrix.  
-- an error. To debug this try: debugmode(true);
```

```
Error in:  
A &= at([x,y],sol[2]); $A ...  
^
```

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

$[0, 0]$

$[a, 0]$

Kemudian 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");
```

```
Variable a1 not found!  
Use global variables or parameters for string evaluation.  
Error in Evaluate, superfluous characters found.  
Try "trace errors" to inspect local variables after errors.  
mxmeval:  
  return evaluate(mxm(s));  
Error in:  
... al("B"),"B"); plotPoint(mxmeval("C"),"C"); plotPoint(mxmeval("A"),"A"): ...  
^
```

Plot segmennya.

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

```
Variable a1 not found!  
Use global variables or parameters for string evaluation.  
Error in Evaluate, superfluous characters found.  
Try "trace errors" to inspect local variables after errors.  
mxmeval:  
  return evaluate(mxm(s));  
Error in:  
plotSegment(mxmeval("A"),mxmeval("C")); plotSegment(mxmeval("B"),mxmeval("C")); pl ...  
^
```

Hitung garis tengah tegak lurus di Maxima.

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

Dan pusat lingkarannya.

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

Kita mendapatkan rumus jari-jari lingkaran luar.

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

$$\frac{\sqrt{a_2^2 + a_1^2} \sqrt{a_2^2 + a_1^2 - 2 a a_1 + a^2}}{2 |a_2|}$$

Mari kita tambahkan ini ke dalam plot.

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

```
Variable a2 not found!  
Use global variables or parameters for string evaluation.  
Error in ^  
Error in expression: [a/2, (a2^2+a1^2-a*a1)/(2*a2)]  
Error in:  
plotPoint(U()); plotCircle(circleWithCenter(mxmeval("U"),mxmeval("distance(U,C)")) ...  
^
```

Dengan menggunakan geometri, kita memperoleh rumus sederhana

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

untuk radius. Kita bisa cek, apakah ini benar adanya pada Maxima. Maxima akan memfaktorkan ini hanya jika kita mengkuadratkannya.

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

$$\left[\frac{a_2^2 + a_1^2}{a_2^2}, 0, \frac{16(a_2^2 + a_1^2)}{a_2^2} \right]$$

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari sembarang segitiga yang tidak sama sisi. Merupakan garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, antara lain ortocenter, sirkumcenter, centroid, titik Exeter dan pusat lingkaran sembilan titik segitiga.

Untuk demonstrasinya, kita menghitung dan memplot garis Euler dalam sebuah segitiga.

Pertama, kita mendefinisikan sudut-sudut segitiga di Euler. Kami menggunakan definisi, yang terlihat dalam ekspresi simbolik.

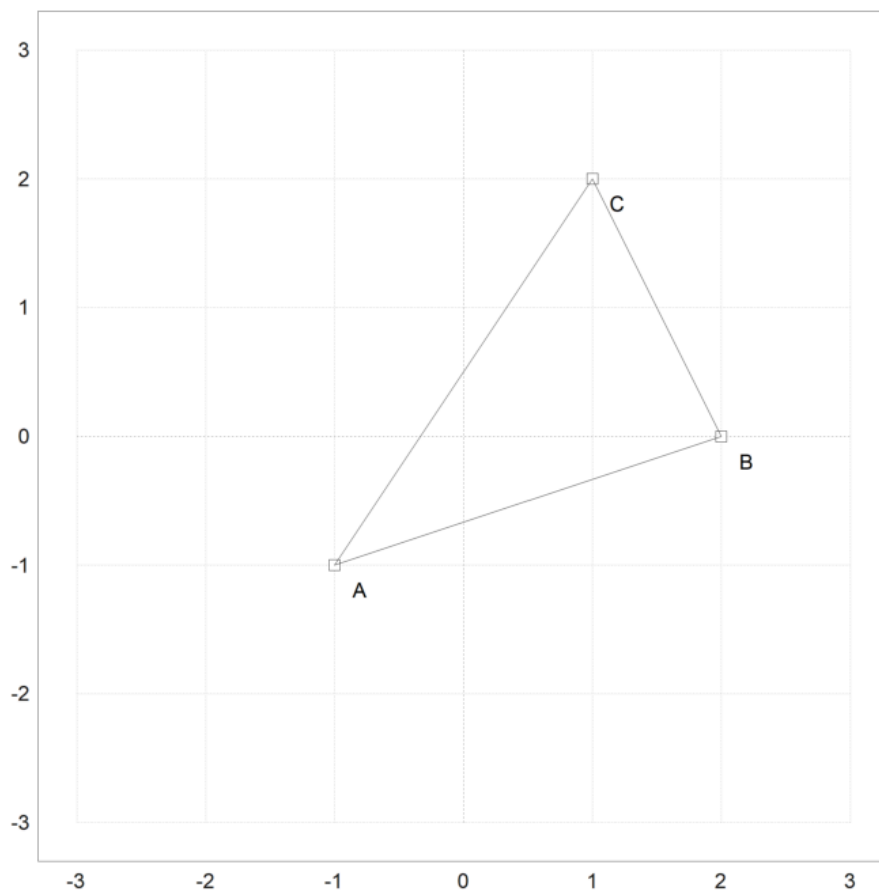
```
>A:=[-1,-1]; B:=[2,0]; C:=[1,2];
```

Untuk memplot objek geometris, kita menyiapkan area plot, dan menambahkan 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 bisa menjumlahkan sisi-sisi segitiga.

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



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

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

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

Kita dapat menghitung koefisien sisi c.

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

```
[- 1, 3, - 2]
```

Dan dapatkan juga rumus untuk baris ini.

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

$$3y - x = -2$$

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

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

$$\frac{3y - x + 2}{\sqrt{10}}$$
$$\frac{7}{\sqrt{10}}$$

Sekarang kita menghitung lingkaran luar ABC.

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

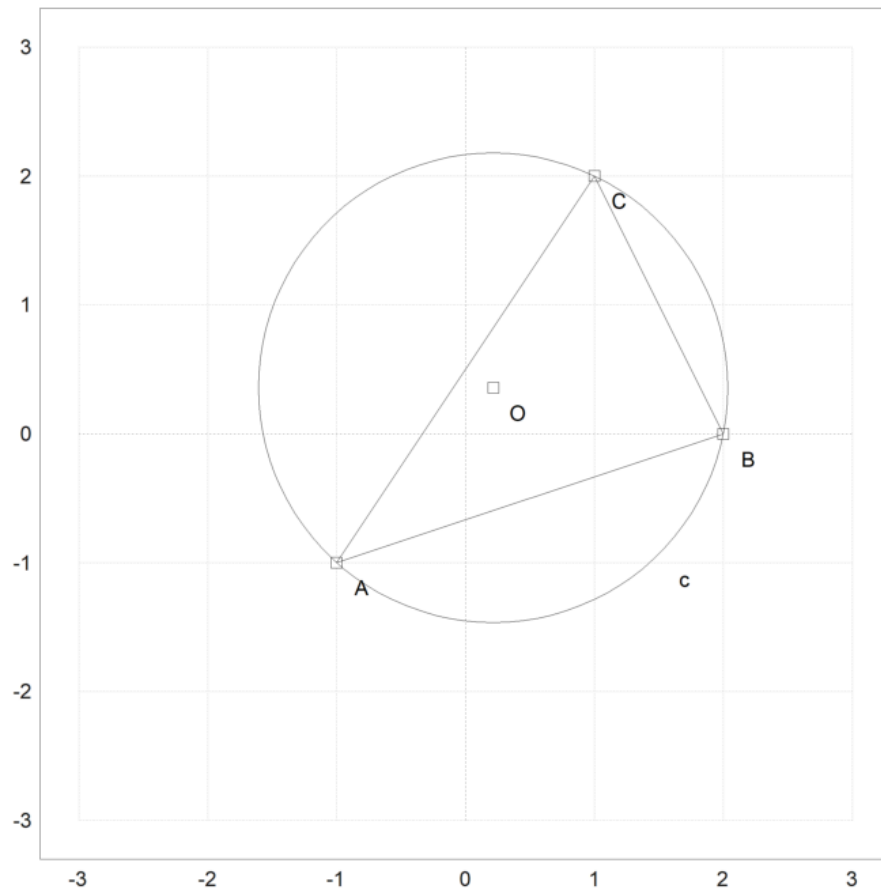
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

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

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

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

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



Kita dapat menghitung perpotongan ketinggian di ABC (ortocenter) secara numerik dengan perintah berikut.

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

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

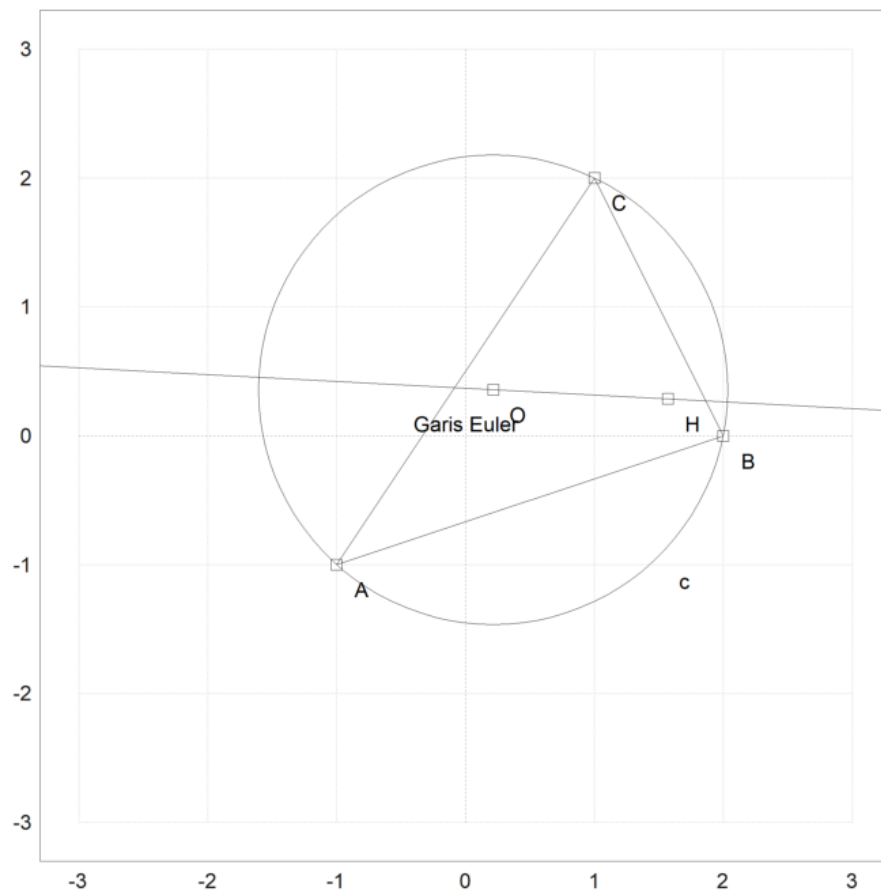
Sekarang kita dapat menghitung garis segitiga Euler.

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

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

Tambahkan ke plot kami.

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

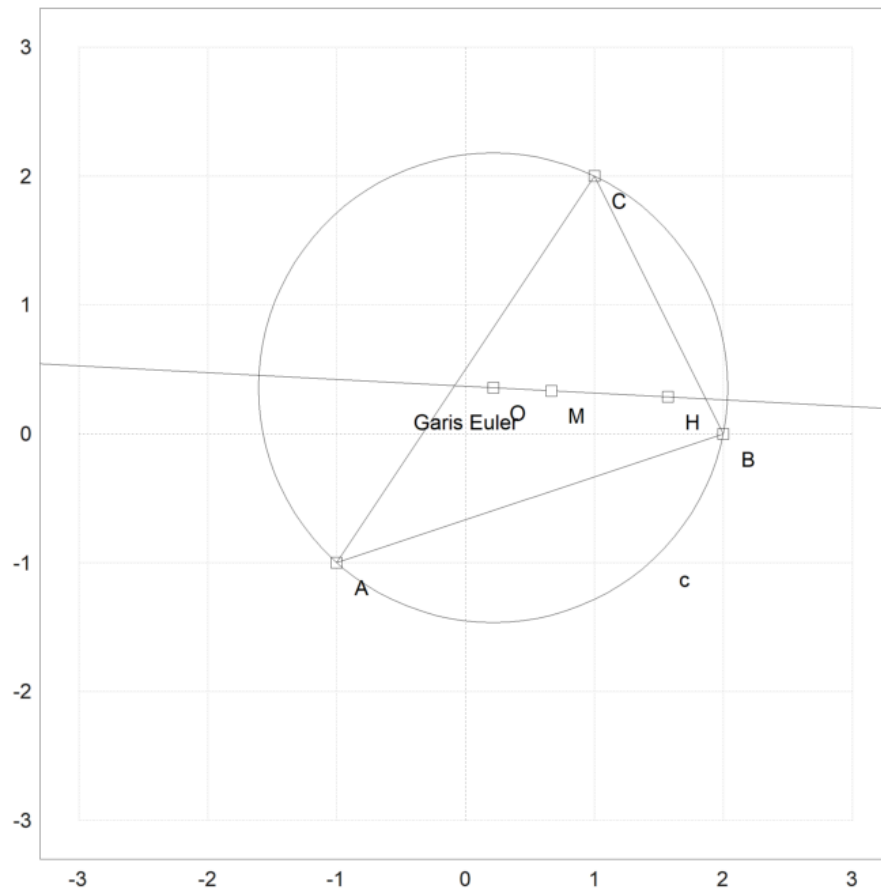


Pusat gravitasi seharusnya berada di garis ini.

```
>M &= (A+B+C)/3; $getLineEquation(e1,x,y) with [x=M[1],y=M[2]]
```

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

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



Teorinya memberitahu kita $MH=2*MO$. Kita perlu menyederhanakan dengan radcan untuk mencapai hal ini.

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

2

Fungsinya mencakup fungsi untuk sudut juga.

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

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

60°15'18.43''

Persamaan pusat 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 jari-jari lingkaran yang tertulis.

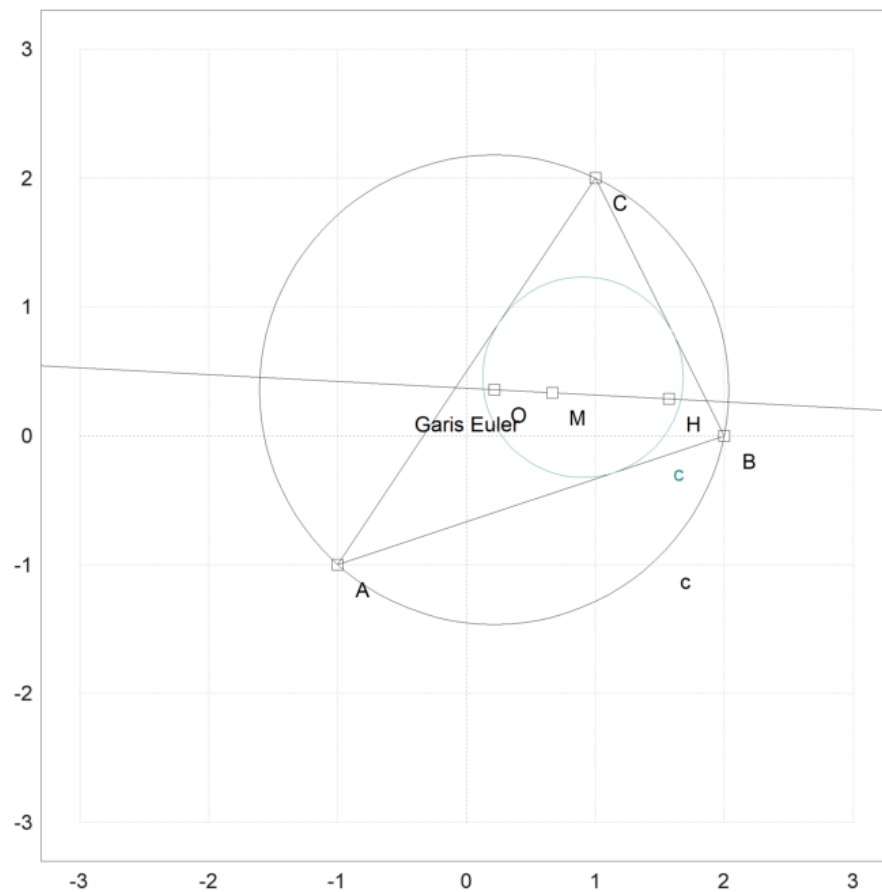
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B))) | ratsimp; $r
```

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

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

Mari kita tambahkan ini ke dalam plot.

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



Parabola

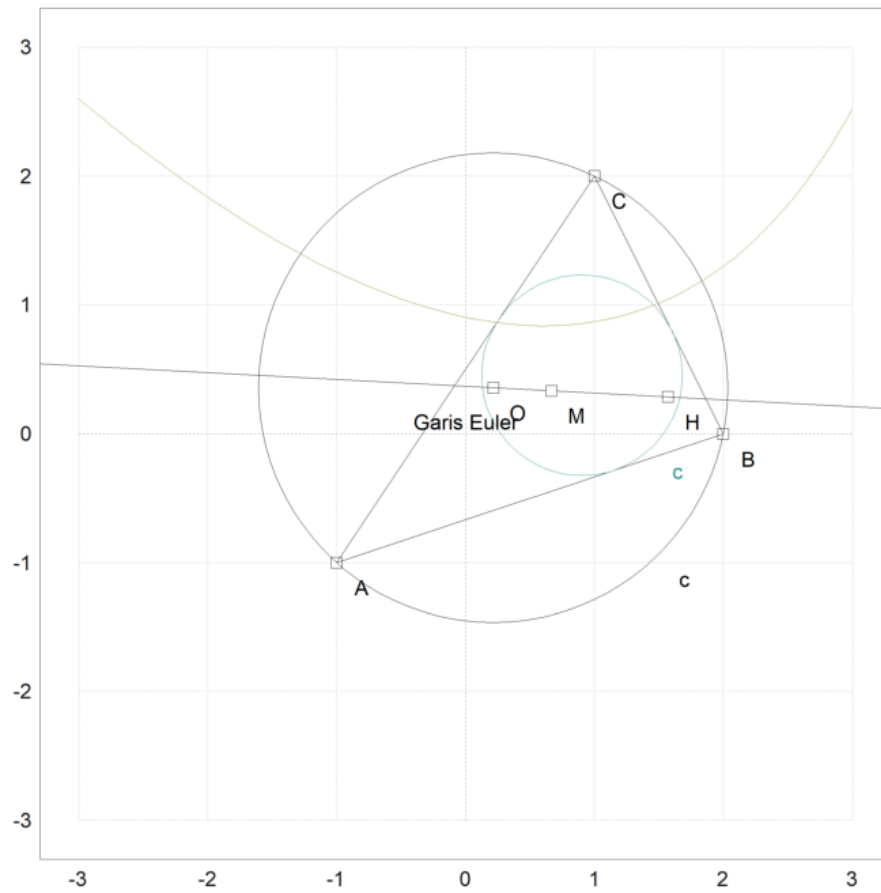
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

```
>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):
```



Ini seharusnya merupakan suatu fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya, jika kita mengkuadratkan persamaannya. Akibatnya, kami mendapatkan solusi palsu.

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

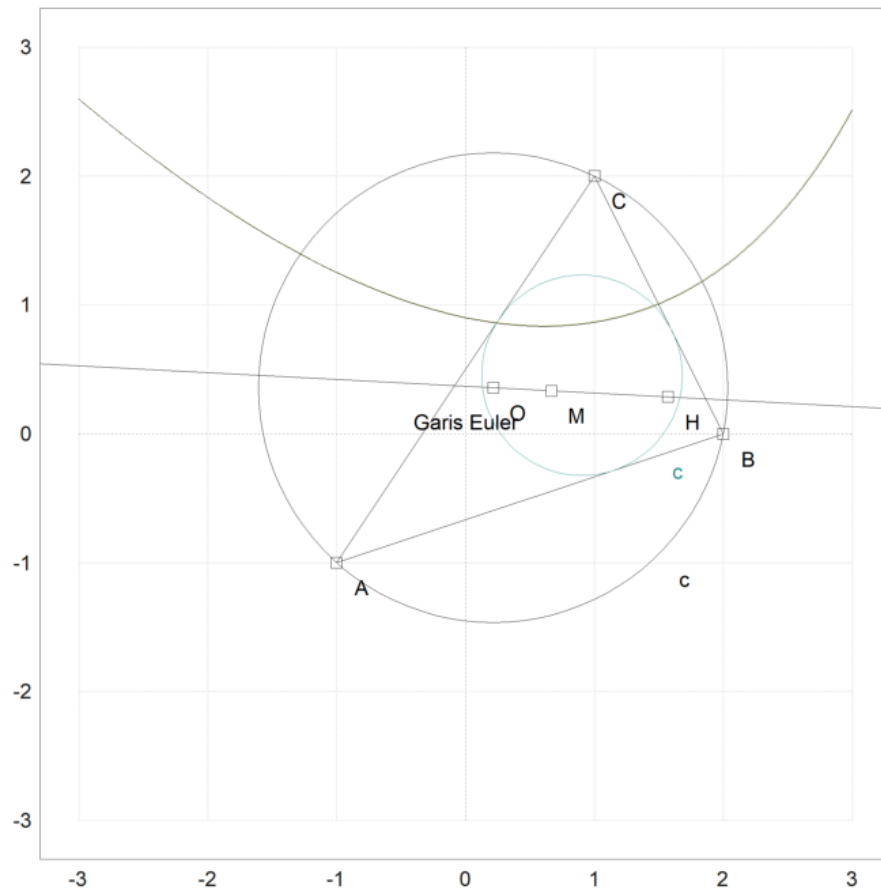
```
[y = - 3 x - sqrt(70) sqrt(9 - 2 x) + 26, y = - 3 x + sqrt(70) sqrt(9 - 2 x) + 26]
```

Solusi pertama adalah

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

Menambahkan solusi pertama pada plot menunjukkan, bahwa itu memang jalan yang kita cari. Teorinya memberitahu kita bahwa itu adalah parabola yang diputar.

```
>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
```

$$\sqrt{1503 - 54 \sqrt{11} \sqrt{70}}$$

$$2.135605779339061$$

```
>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
```

$$\sqrt{1503 - 54 \sqrt{11} \sqrt{70}}$$

$$2.135605779339061$$

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

Hal ini terinspirasi dari ceramah N.J.Wildberger. Dalam bukunya "Divine Proportions", Wildberger mengusulkan untuk mengganti gagasan klasik tentang jarak dan sudut dengan kuadran dan penyebaran. Dengan menggunakan hal ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap "rasional".

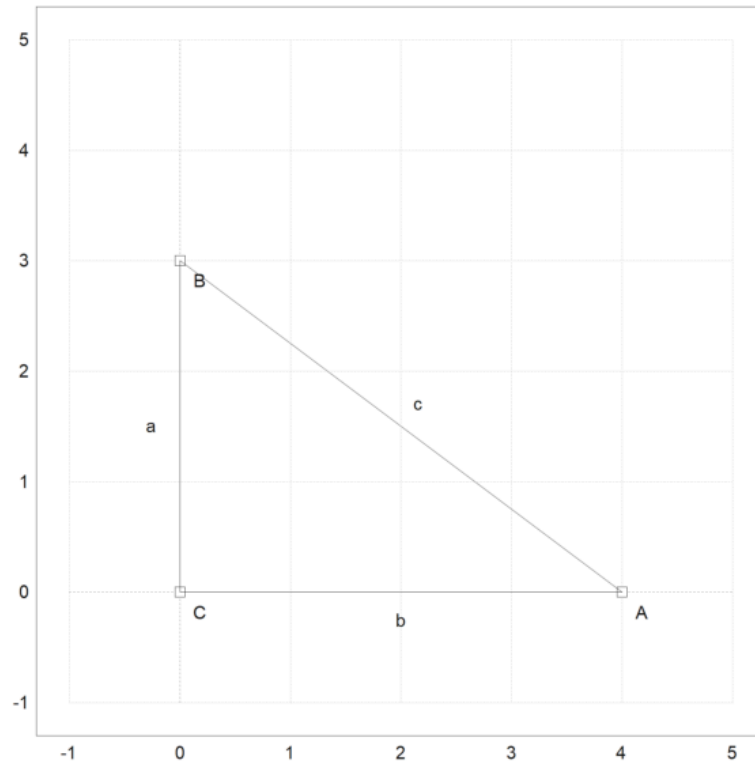
Berikut ini, saya memperkenalkan konsep, dan memecahkan beberapa masalah. Saya menggunakan perhitungan simbolik Maxima di sini, yang menyembunyikan keunggulan utama trigonometri rasional yaitu perhitungan hanya dapat dilakukan dengan kertas dan pensil. Anda diundang untuk memeriksa hasilnya tanpa komputer.

Intinya adalah perhitungan rasional simbolik seringkali memberikan hasil yang sederhana. Sebaliknya, trigonometri klasik menghasilkan hasil trigonometri yang rumit, yang hanya mengevaluasi perkiraan numerik saja.


```
>load geometry;
```

For a first introduction, we use the rectangular triangle with the famous Egyptian proportions 3, 4 and 5. The following commands are Euler commands to plot plane geometry contained in the Euler file "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,

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

dimana w_a adalah sudut di A. Cara umum untuk menghitung sudut ini adalah dengan mengambil invers dari fungsi sinus. Hasilnya adalah sudut yang tidak dapat dicerna, yang hanya dapat dicetak secara kasar.

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

```
36°52'11.63''
```

Trigonometri rasional mencoba menghindari hal ini.

Gagasan pertama tentang trigonometri rasional adalah kuadran, yang menggantikan jarak. Faktanya, itu hanyalah jarak yang dikuadratkan. Di bawah ini, a, b, dan c menyatakan kuadran sisi-sisinya.

Teorema Pythagoras menjadi $a+b=c$.

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

```
25 = 25
```

Pengertian trigonometri rasional yang kedua adalah penyebaran. Penyebaran mengukur pembukaan antar garis. Nilainya 0 jika garisnya sejajar, dan 1 jika garisnya persegi panjang. Ini adalah kuadrat sinus sudut antara dua garis.

Luas garis AB dan AC pada gambar di atas didefinisikan sebagai

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

dimana a dan c adalah kuadran suatu segitiga siku-siku yang salah satu sudutnya berada di A.

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

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

Tentu saja ini lebih mudah dihitung daripada sudutnya. Namun Anda kehilangan properti bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita dapat mengonversi nilai perkiraan sudut wa menjadi sprad, dan mencetaknya sebagai pecahan.

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

$$9/25$$

Hukum kosinus trgonometri klasik diterjemahkan menjadi "cross law" berikut.

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

Di sini a, b, dan c adalah kuadran sisi-sisi segitiga, dan sa adalah jarak di sudut A. Sisi a, seperti biasa, berhadapan dengan sudut A.

Hukum-hukum ini diterapkan dalam file geometri.e yang kami muat ke Euler.

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

$$\left[\left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{5}{3\sqrt{2}} \right)^2 \right] = \left[\frac{14 bb (1 - saa)}{3}, \frac{14 bb (1 - saa)}{3}, \frac{5 2^{\frac{3}{2}} bb (1 - saa)}{3} \right]$$

Dalam kasus kami, kami mendapatkan

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

$$1024 = 1024$$

Mari kita gunakan hukum silang ini untuk mencari penyebaran di A. Untuk melakukannya, kita buat hukum silang untuk kuadran a, b, dan c, dan selesaikan untuk penyebaran yang tidak diketahui sa.

Anda bisa melakukannya dengan tangan dengan mudah, tapi saya menggunakan Maxima. Tentu saja, kami mendapatkan hasilnya, kami sudah mendapatkannya.

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

$$1024 = 1600 (1 - x)$$

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

Kami sudah mengetahui hal ini. Pengertian penyebaran merupakan kasus khusus dari hukum silang.

Kita juga dapat menyelesaikannya untuk persamaan umum a,b,c. Hasilnya adalah rumus yang menghitung penyebaran sudut suatu segitiga berdasarkan kuadran ketiga sisinya.

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

$$\left[\left[\frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36}, \frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36}, \frac{15 2^{\frac{3}{2}} bb x + 18 bb^2 + (-36 aa - 15 2^{\frac{1}{2}}) bb + 18 aa^2 - 15 2^{\frac{1}{2}} aa + 25}{18} \right] = 0 \right]$$

Kita bisa membuat fungsi dari hasilnya. Fungsi seperti itu sudah didefinisikan dalam file geometri.e Euler.

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

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

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

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

Hasilnya rasional, yang tidak mudah didapat jika kita menggunakan trigonometri klasik.

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

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

Ini adalah sudut dalam derajat.

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

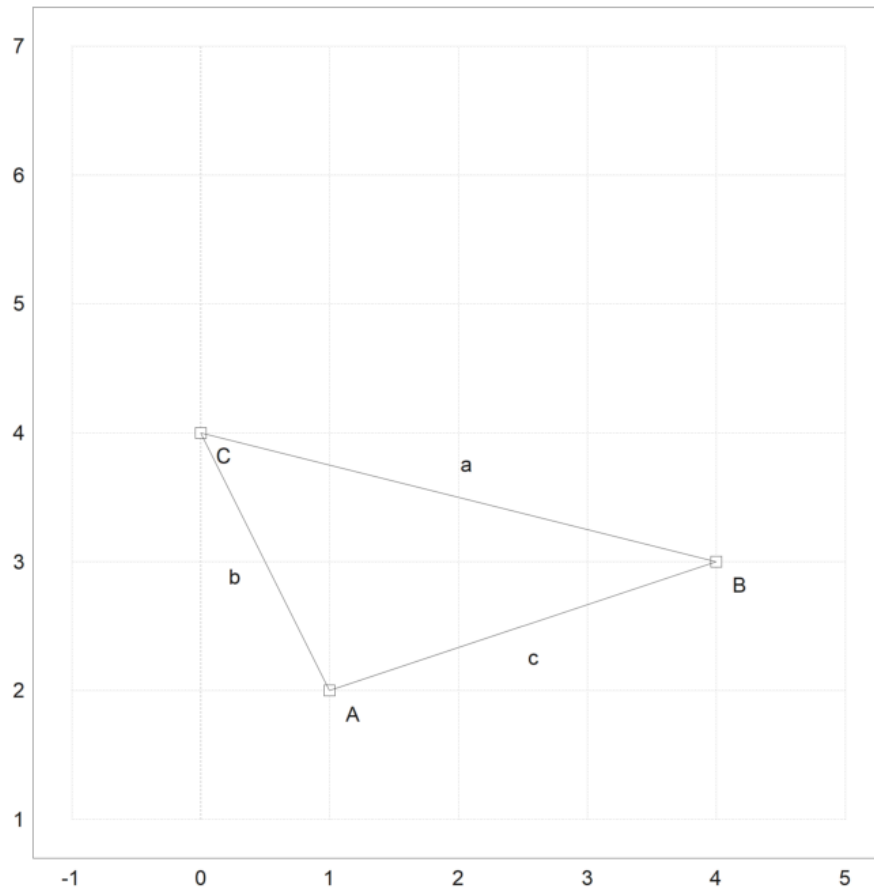
$$67^{\circ}47'32.44''$$

Contoh Lain

Sekarang, mari kita coba contoh lebih lanjut.

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

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

$$\sqrt{10}$$

Euler juga memuat fungsi kuadran antara dua titik.

Pada contoh berikut, karena $c+b$ bukan a , maka segitiga tersebut bukan persegi panjang.

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

$$10$$

$$5$$

$$17$$

Pertama, mari kita hitung sudut tradisional. Fungsi computeAngle menggunakan metode biasa berdasarkan perkalian titik dua vektor. Hasilnya adalah beberapa perkiraan floating point.

$$\begin{aligned} A &= \langle 1, 2 \rangle \quad B = \langle 4, 3 \rangle, \quad C = \langle 0, 4 \rangle \\ \mathbf{a} &= C - B = \langle -4, 1 \rangle, \quad \mathbf{c} = A - B = \langle -3, -1 \rangle, \quad \beta = \angle ABC \\ \mathbf{a} \cdot \mathbf{c} &= |\mathbf{a}| \cdot |\mathbf{c}| \cos \beta \\ \cos \angle ABC = \cos \beta &= \frac{\mathbf{a} \cdot \mathbf{c}}{|\mathbf{a}| \cdot |\mathbf{c}|} = \frac{12 - 1}{\sqrt{17} \sqrt{10}} = \frac{11}{\sqrt{17} \sqrt{10}} \end{aligned}$$

```
>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. Kita masukkan kuadran a, b, dan c ke dalam hukum silang dan selesaikan x.

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

$$4 = 200(1 - x)$$

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

Yaitu, fungsi penyebaran yang didefinisikan dalam "geometri.e".

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

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

Maxima mendapatkan hasil yang sama dengan menggunakan trigonometri biasa, jika kita memaksakannya. Itu menyelesaikan suku $\sin(\arccos(\dots))$ menjadi hasil pecahan. Kebanyakan siswa tidak dapat melakukan hal ini.

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

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

Setelah kita mendapatkan sebaran di B, kita dapat menghitung tinggi ha pada sisi a. Ingat itu

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

menurut definisi.

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

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

Gambar berikut dihasilkan dengan program geometri C.a.R., yang dapat menggambar kuadran dan sebaran.

image: (20) Rational_Geometry_CaR.png

Menurut definisi, panjang ha adalah akar kuadrat dari kuadrannya.

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

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

Sekarang kita dapat menghitung luas segitiga tersebut. Jangan lupa, bahwa kita sedang berhadapan dengan kuadran!

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

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

Rumus 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-tama kita menghitung penyebaran di B untuk sebuah segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat ("quadrea"?), memfaktorkannya dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

Memang benar, hal ini sulit dilakukan dengan pensil dan kertas.

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

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

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

Aturan Penyebaran Tiga Kali Lipat

Kerugian dari spread adalah bahwa mereka tidak lagi hanya menambahkan sudut yang sama.

Namun, tiga spread 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 tiga sudut mana pun yang besarnya 180°.

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

Sejak menyebarnya

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

sama, aturan penyebaran tiga kali lipat juga benar, jika

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

Karena penyebaran sudut negatifnya sama, maka aturan penyebaran tiga kali lipat juga berlaku, jika

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

Misalnya, kita dapat menghitung penyebaran sudut 60°. Ini 3/4. Namun persamaan tersebut memiliki solusi kedua, dimana semua spread adalah 0.

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

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

Penyebaran 90° jelas sama dengan 1. Jika dua sudut dijumlahkan menjadi 90°, penyebarannya menyelesaikan persamaan penyebaran rangkap tiga dengan a,b,1. Dengan perhitungan berikut kita mendapatkan a+b=1.

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

$$(y + x + 1)^2 = 2(y^2 + x^2 + 1) + 4xy$$

$$[x = 1 - y]$$

Karena penyebaran 180°-t sama dengan penyebaran t, rumus penyebaran tiga kali lipat juga berlaku, jika salah satu sudut adalah jumlah atau selisih dua sudut lainnya.

Sehingga kita dapat mencari penyebaran sudut dua kali lipat tersebut. Perhatikan bahwa ada dua solusi lagi. Kami menjadikan ini sebuah fungsi.

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

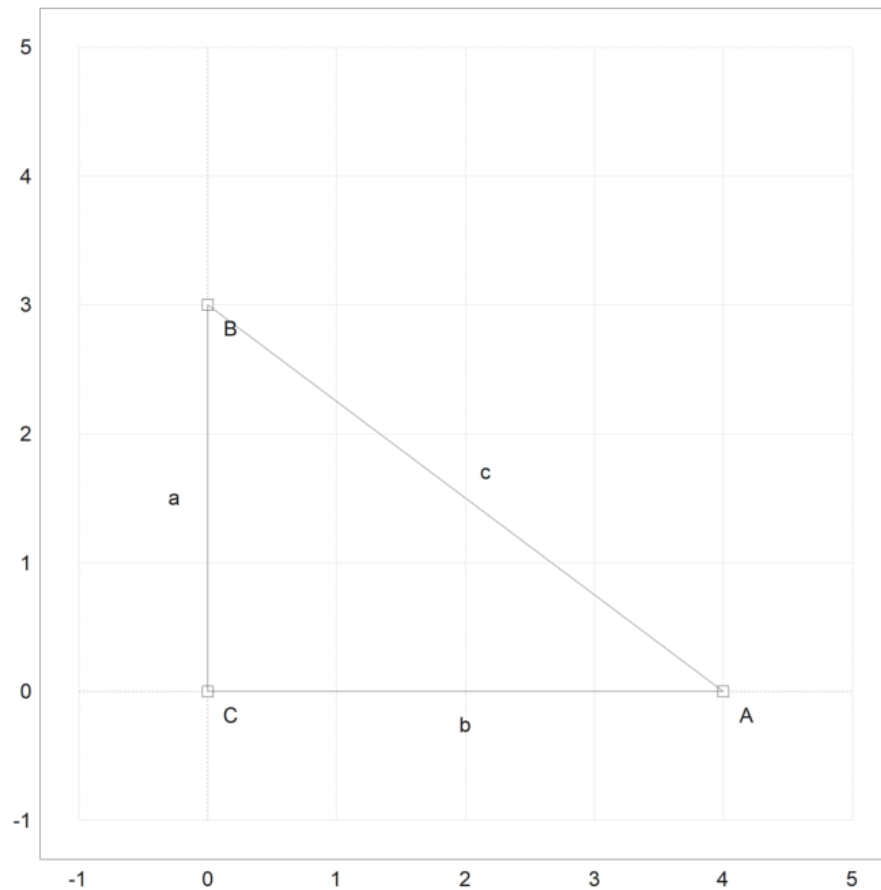
$$[x = 4a - 4a^2, x = 0]$$

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

Pembagi Sudut

Inilah 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. Namun kita ingin menyelesaikannya secara umum a,b,c.

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

Jadi pertama-tama kita menghitung penyebaran sudut yang dibagi dua di A, menggunakan rumus penyebaran tiga kali lipat.

Masalah dengan rumus ini muncul lagi. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu pada sudut membagi dua 180° -wa.

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

$$\left(2x + \frac{a}{b+a}\right)^2 = 2 \left(2x^2 + \frac{a^2}{(b+a)^2}\right) + \frac{4ax^2}{b+a}$$

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

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

Mari kita periksa untuk persegi panjang Mesir.

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

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

Kita dapat mencetak sudut dalam Euler, setelah mentransfer penyebarannya ke radian.

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

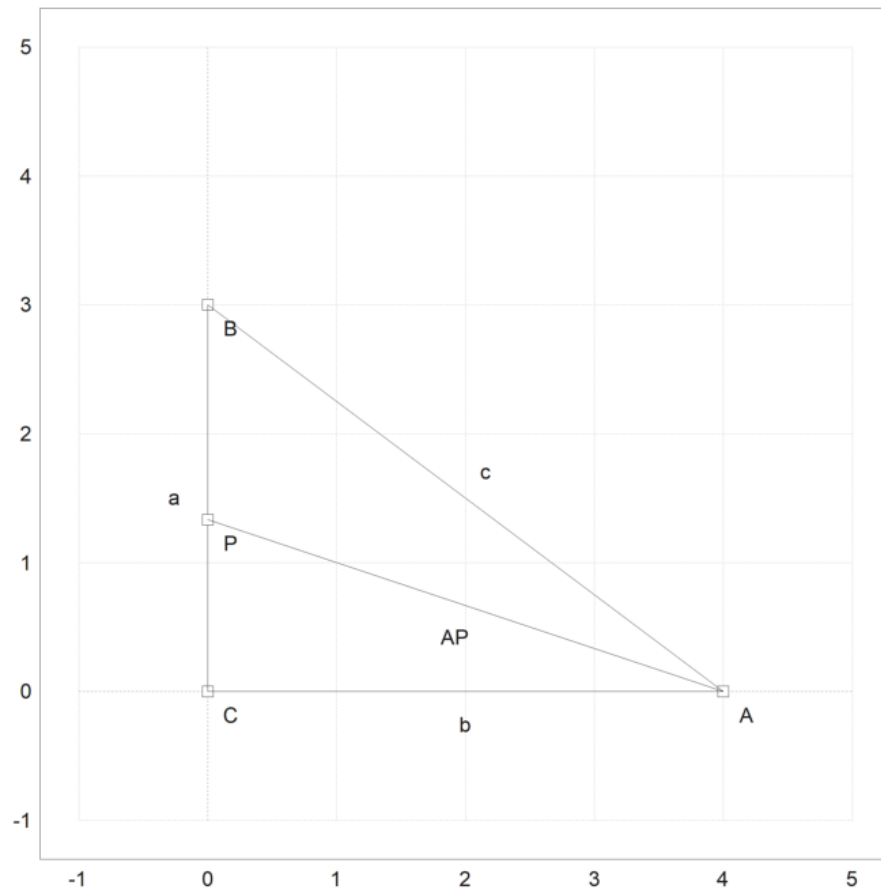
```
18°26'5.82''
```

Titik P merupakan 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 menghitung panjang garis bagi AP.

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

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

berlaku di segitiga mana pun. Jika digabungkan, maka hal ini akan diterjemahkan ke dalam apa yang disebut dengan "spread law"

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

dimana a,b,c menunjukkan qudrance.

Karena spread CPA adalah $1-sa^2$, kita memperolehnya bisa $1/b/(1-sa^2)$ dan dapat menghitung bisa (kuadran dari garis bagi 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 Egyptian kita.

```
>sqr(mxeval("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 peroleh dengan rumus trigonometri.

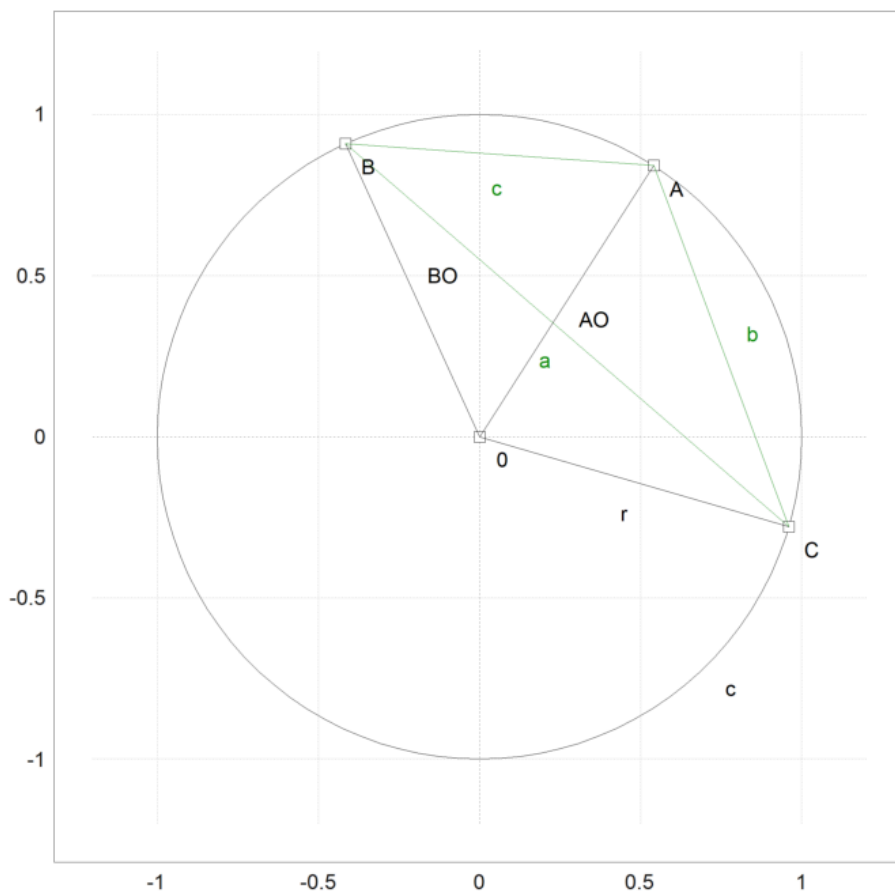
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

1.33333333333

Sudut Akord

Lihatlah 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 di pusat O untuk r. Jadi kita mendapatkan rumus jari-jari kuadrat dari perilingkaran dalam kuadran sisi-sisinya.

Kali ini, Maxima menghasilkan beberapa angka 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
```

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

Kita dapat menjadikannya fungsi Euler.

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

Mari kita periksa hasil untuk poin kita 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, penyebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut tali busur.

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

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

Faktanya, penyebarannya adalah $b/(4r)$, dan kita melihat bahwa sudut tali busur b adalah setengah sudut pusatnya.

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

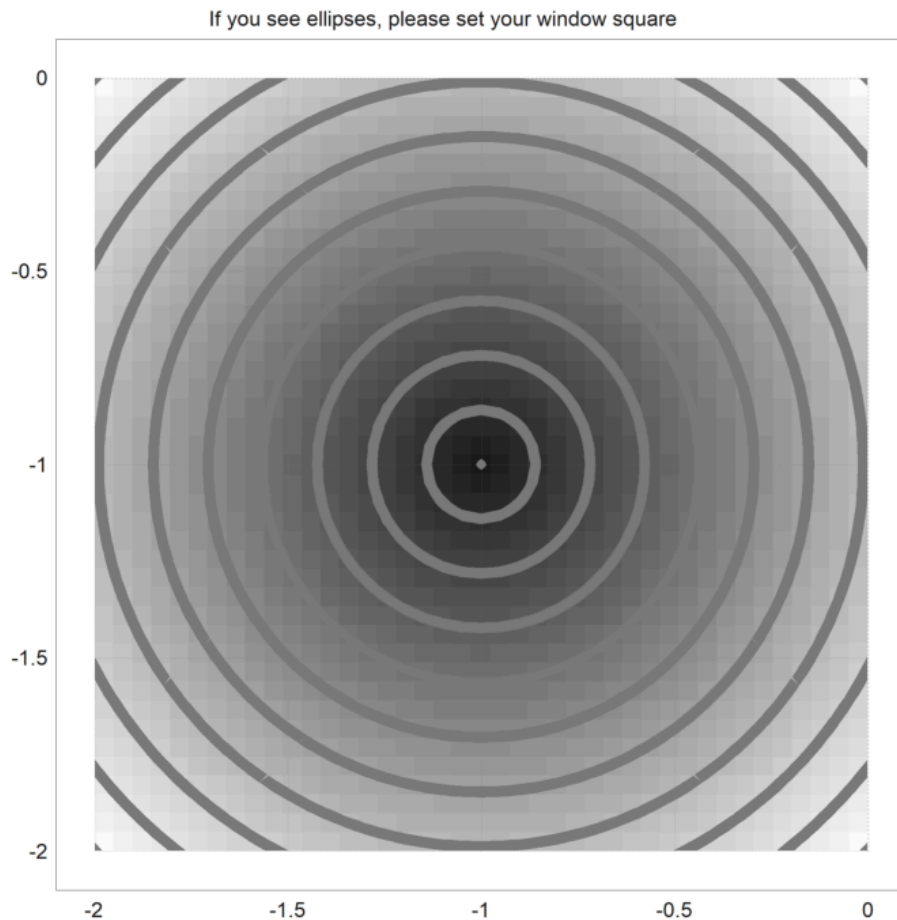
0

Contoh 6: Jarak Minimal pada Bidang

Catatan awal

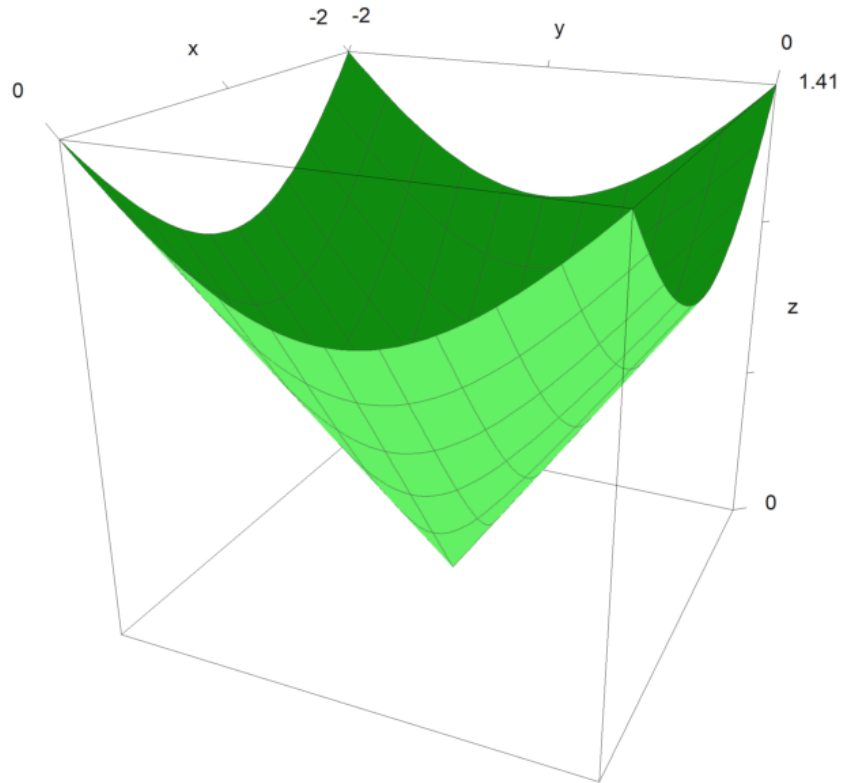
Fungsi yang, ke titik M pada bidang, menetapkan jarak AM antara titik tetap A dan M, mempunyai garis datar yang cukup sederhana: lingkaran berpusat di A.

```
>&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 cukup sederhana: bagian atas kerucut:

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

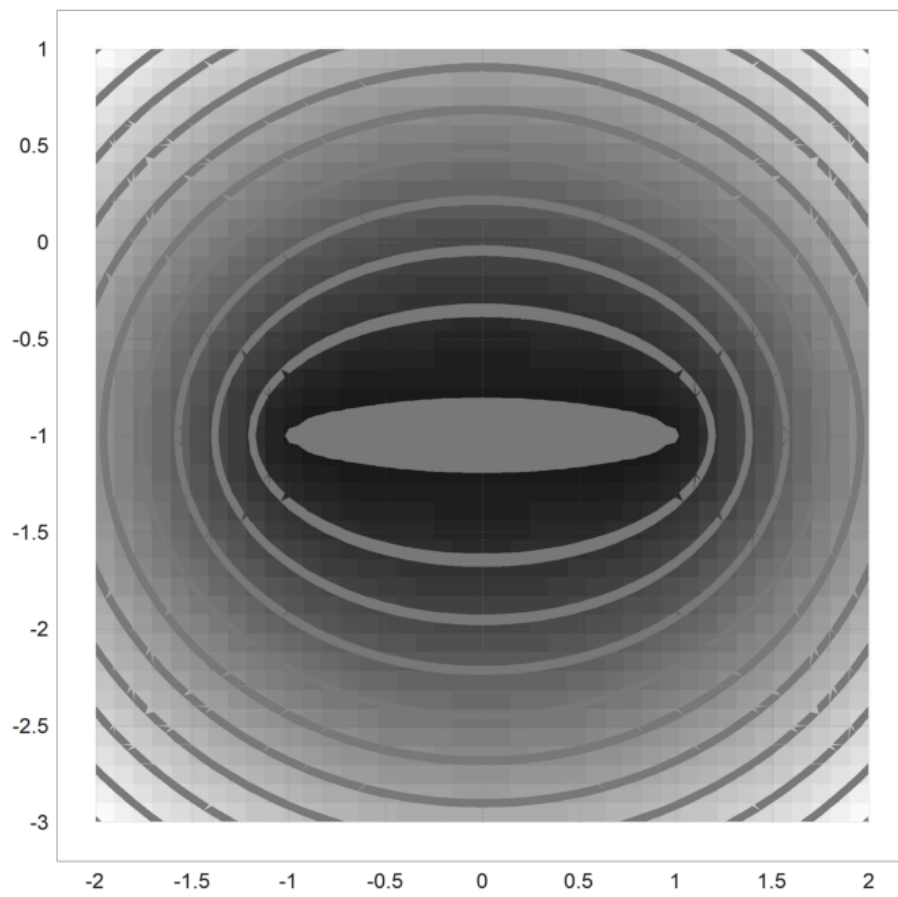


Tentu saja minimum 0 dicapai di A.

Dua Titik

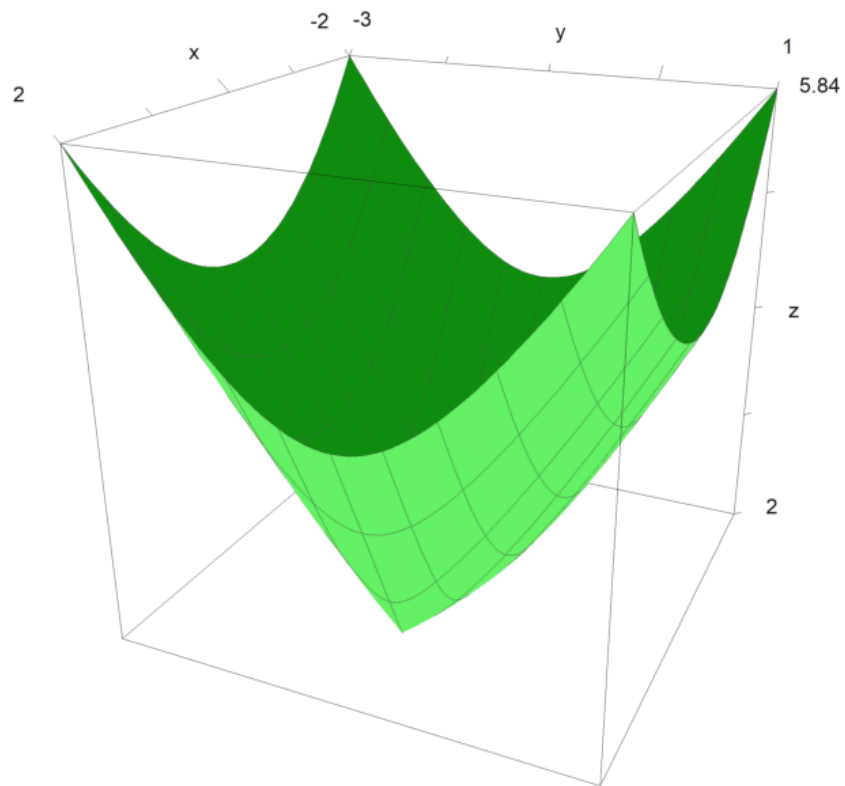
Sekarang kita lihat fungsi $MA+MB$ dimana A dan B adalah dua titik (tetap). Merupakan "fakta yang diketahui" bahwa kurva tingkat berbentuk elips, titik fokusnya adalah A dan B; kecuali AB minimum yang konstan pada ruas $[AB]$:

```
>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):
```



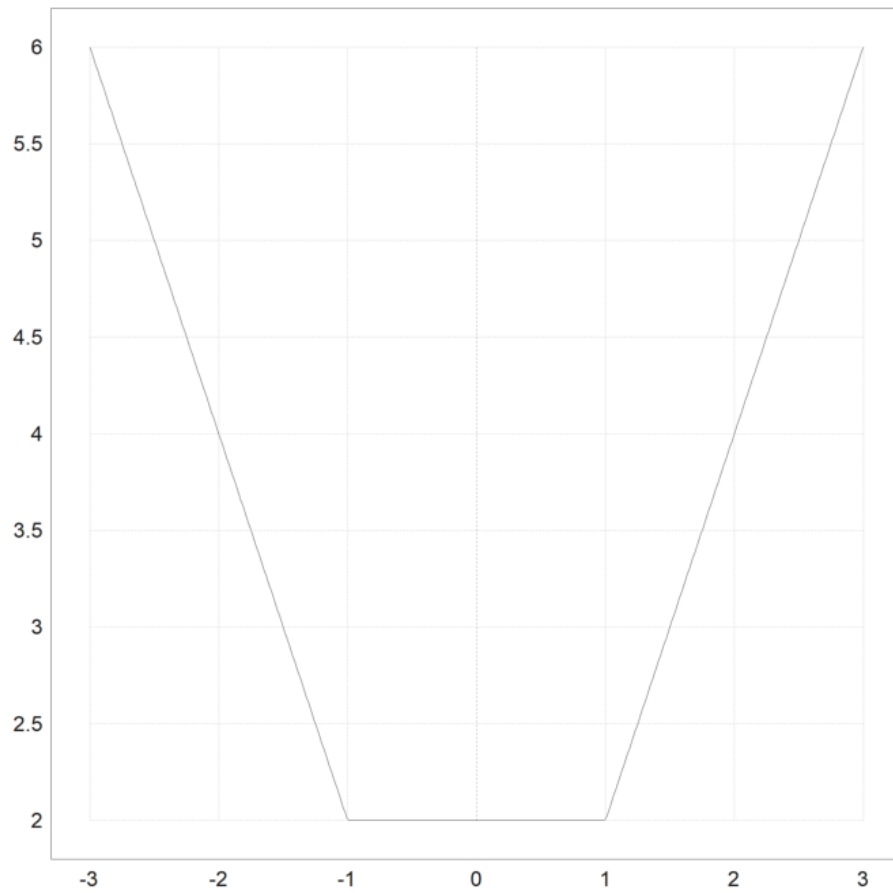
Grafiknya lebih menarik:

```
>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):
```



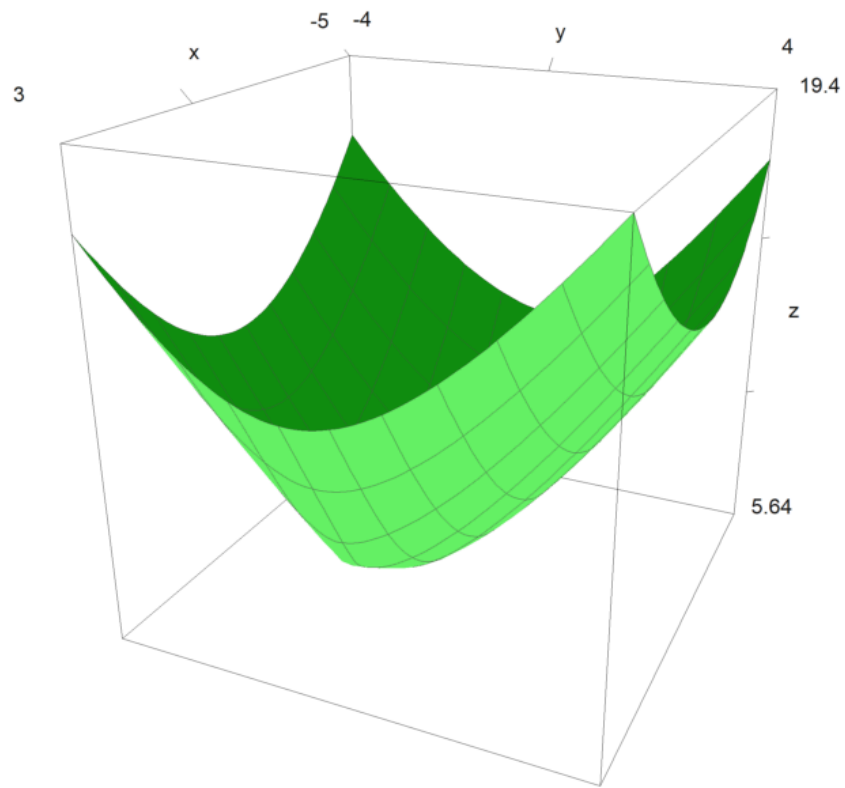
Tiga Titik

Kini segalanya menjadi lebih sederhana: Tidak diketahui secara luas bahwa $MA+MB+MC$ mencapai nilai minimumnya pada satu titik pada bidang tersebut, namun untuk menentukannya tidaklah mudah:

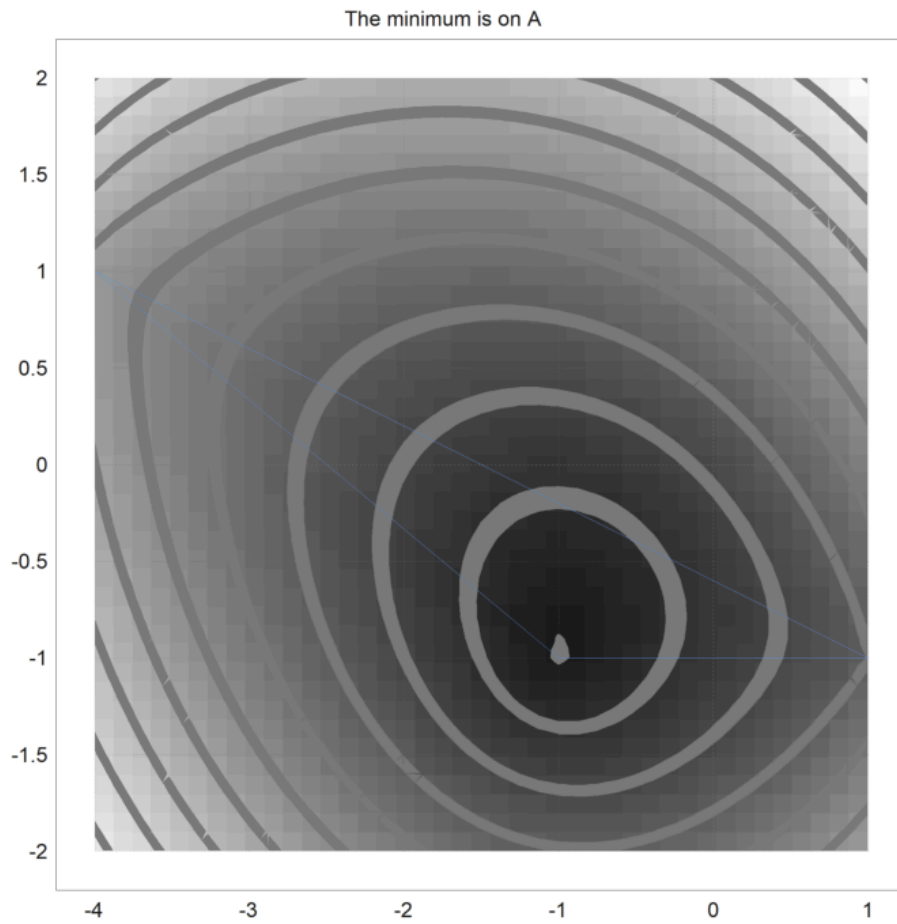
1) Jika salah satu sudut segitiga ABC lebih dari 120° (katakanlah di A), maka sudut minimum dicapai pada titik tersebut (katakanlah $AB+AC$).

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;
```

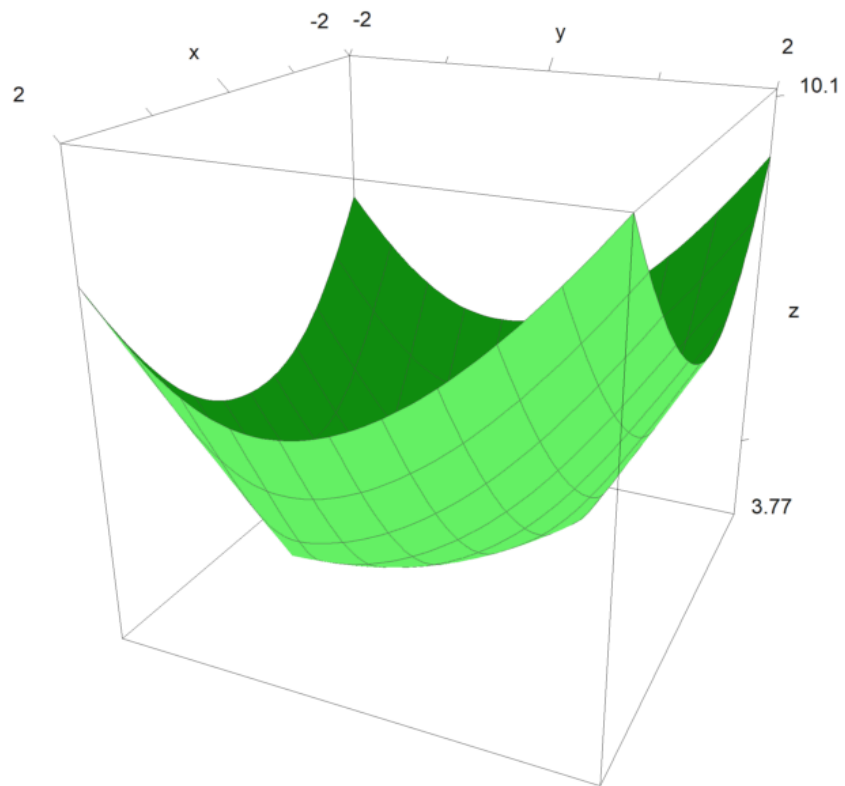


```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");  
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);  
>insimg;
```

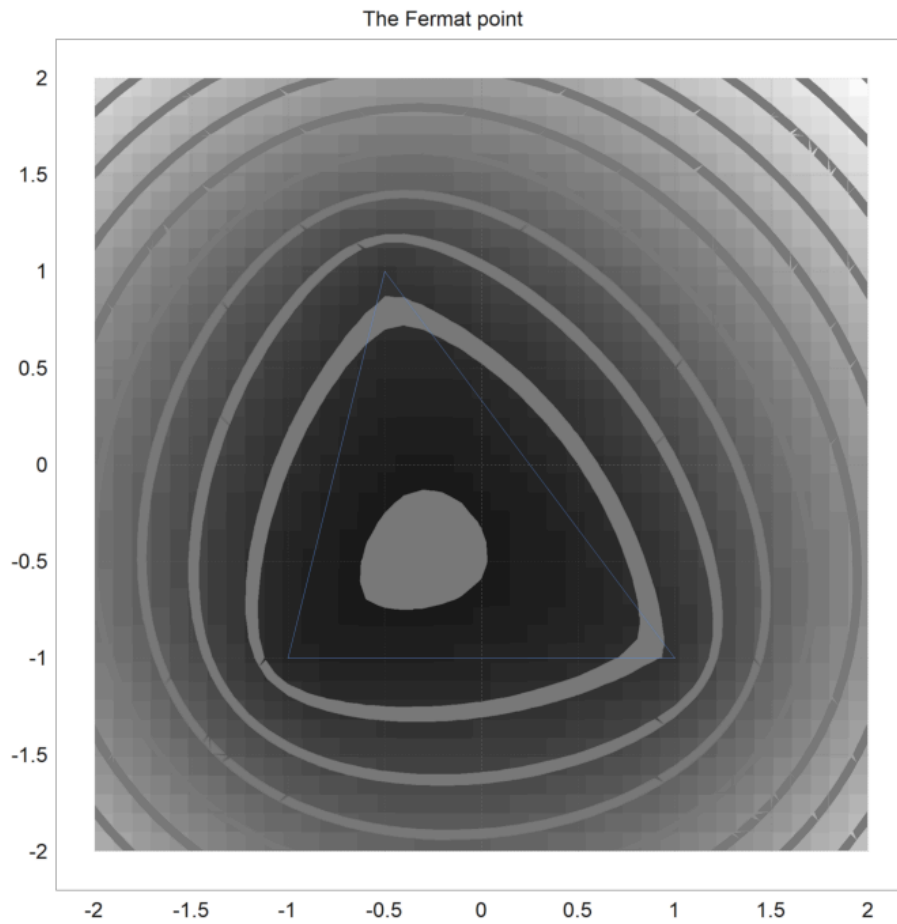


2) Tetapi jika semua sudut segitiga ABC kurang dari 120° , maka titik minimum ada di titik F di bagian dalam segitiga, yaitu satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (maka masing-masing sudutnya 120°):

```
>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;
```

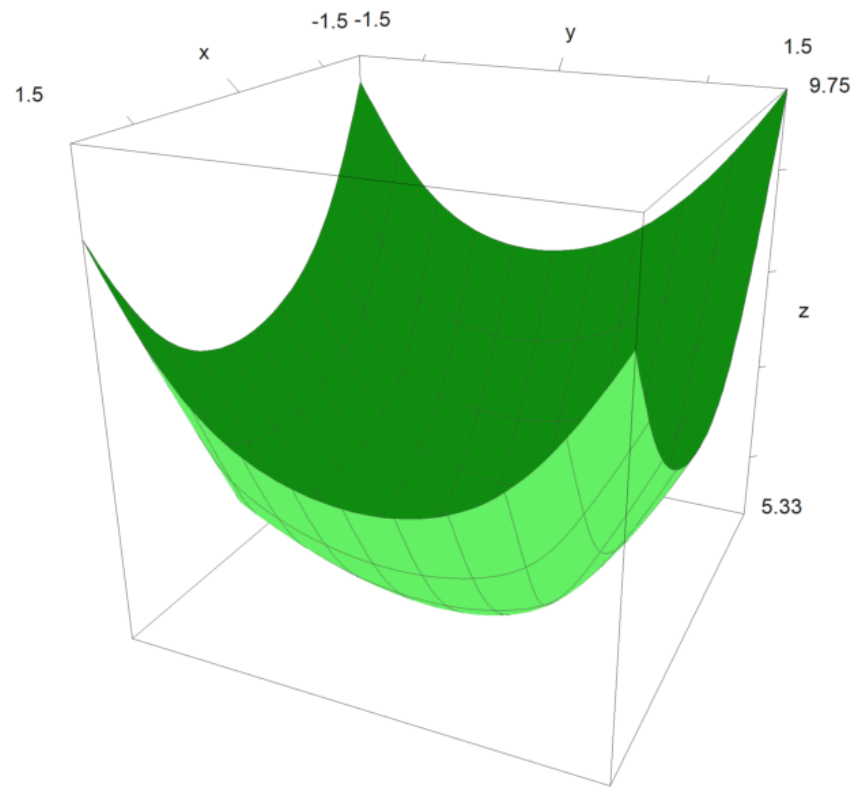
Merupakan kegiatan yang menarik untuk merealisasikan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu soft tertulis di Java yang memiliki instruksi "garis kontur"...

Semua hal di atas ditemukan oleh seorang hakim Perancis bernama Pierre de Fermat; dia menulis surat kepada para penggila lainnya seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di bagian pajak penghasilan. Jadi titik unik F sehingga $FA+FB+FC$ minimal disebut titik Fermat segitiga. Namun nampaknya beberapa tahun sebelumnya, Torricelli dari Italia telah menemukan titik ini sebelum Fermat menemukannya! Pokoknya tradisinya adalah memperhatikan hal ini F...

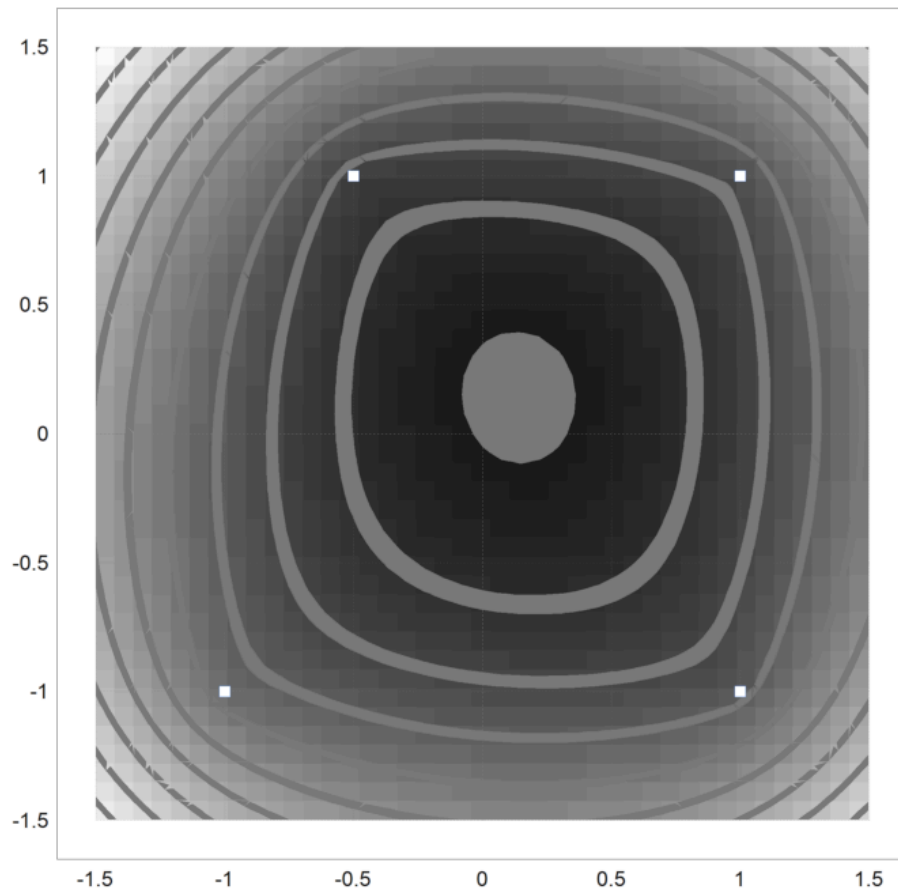
Empat Titik

Langkah selanjutnya adalah menambahkan poin ke-4 D dan mencoba meminimalkan $MA+MB+MC+MD$; katakanlah Anda seorang operator TV kabel dan ingin mencari di bidang mana Anda harus memasang antena sehingga Anda dapat memberi makan empat desa dan menggunakan 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 nilai minimum dan tidak tercapai di simpul A, B, C, atau D:

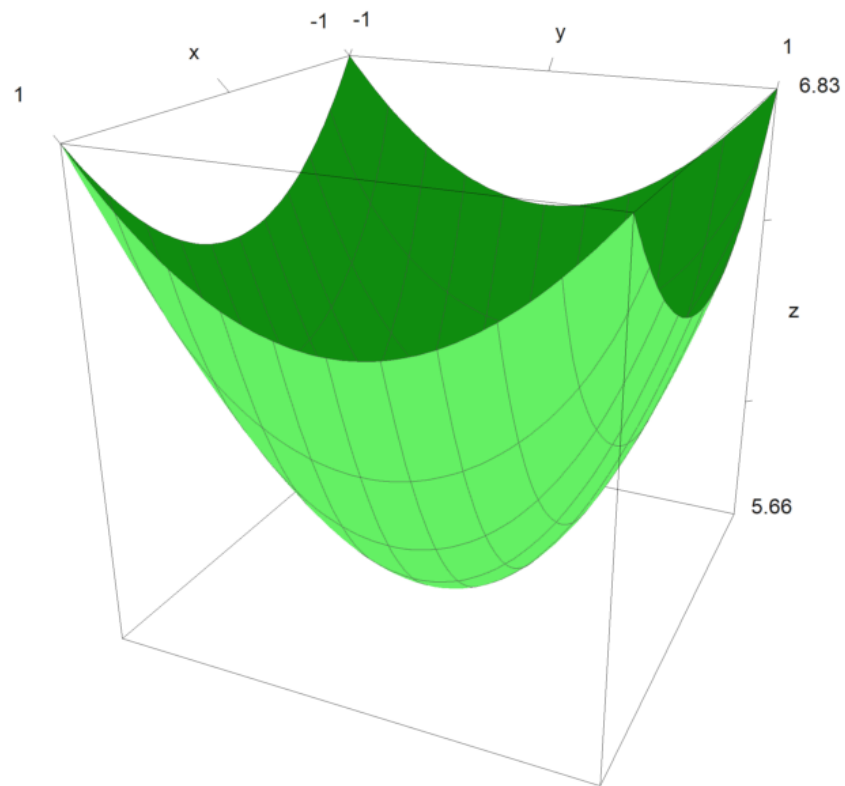
```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```

```
[0.142858, 0.142857]
```

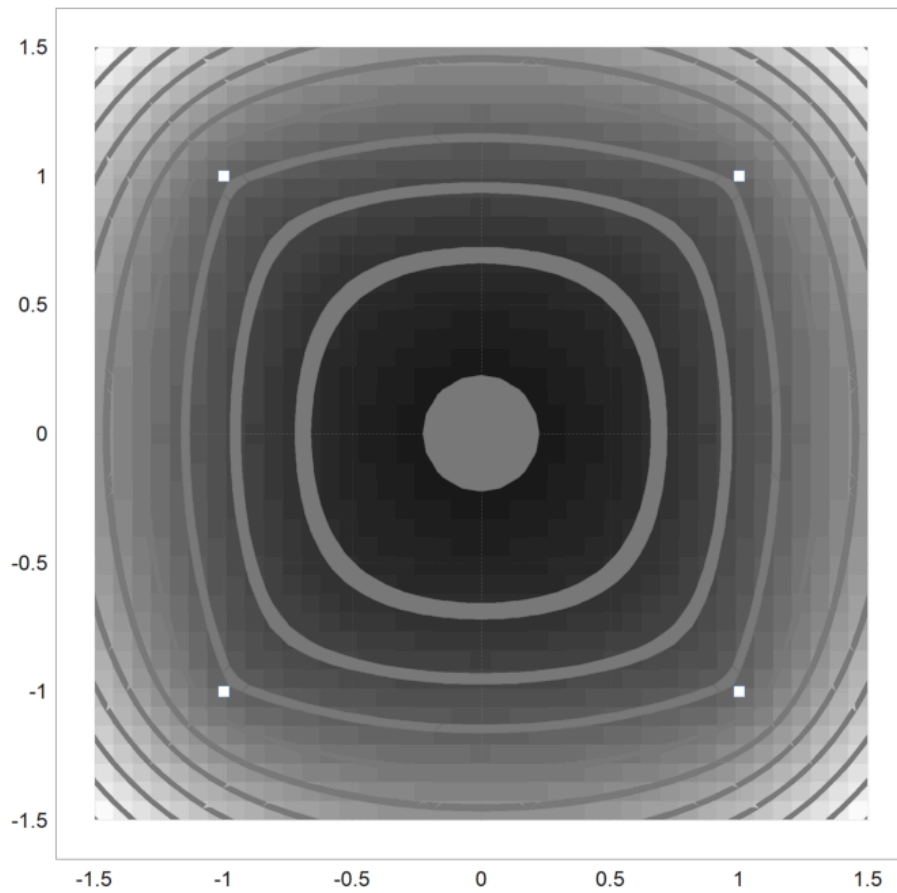
Nampaknya dalam hal ini koordinat titik optimal bersifat rasional atau mendekati rasional...

Sekarang ABCD adalah persegi, kita berharap titik optimalnya adalah 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 diperhatikan gambar di bawah, terlihat bahwa kita membutuhkan dua lingkaran yang menyentuh dua garis yang membentuk kerucut, dan satu garis yang membentuk bidang yang memotong kerucut.

Kami menggunakan file geometry.e Euler untuk ini.

```
>load geometry;
```

Pertama dua garis membentuk kerucut.

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

```
[- a, 1, 0]
```

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

```
[- a, - 1, 0]
```

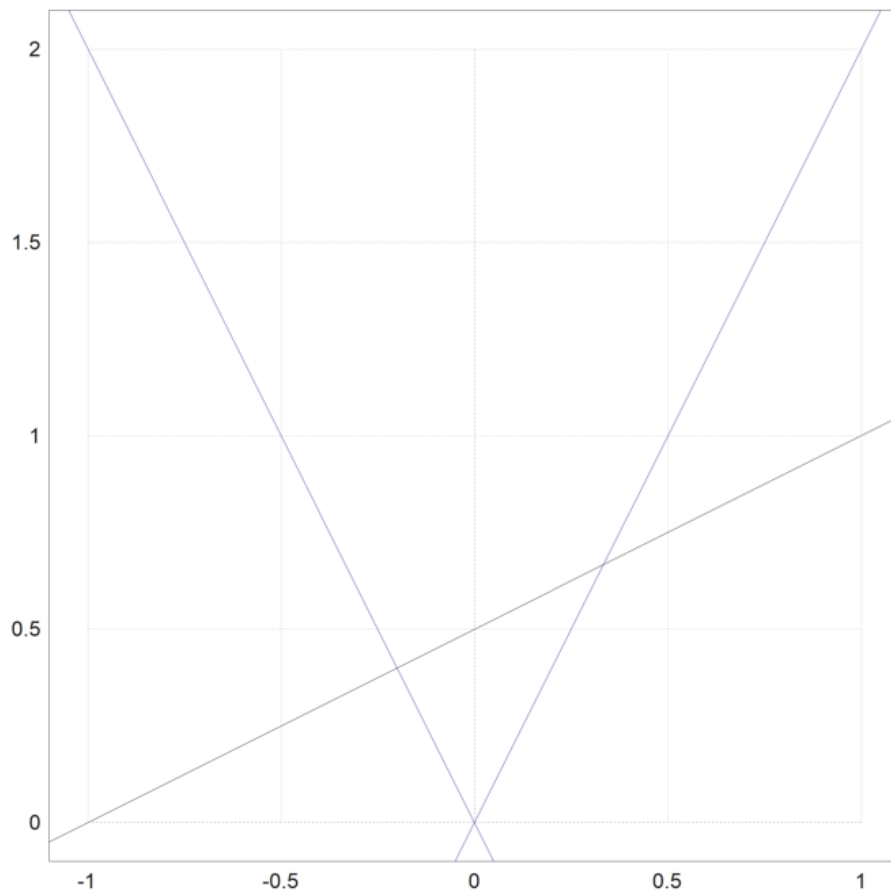
Lalu baris ketiga.

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

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

Kami 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 g1.

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

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

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 tentukan 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.

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

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

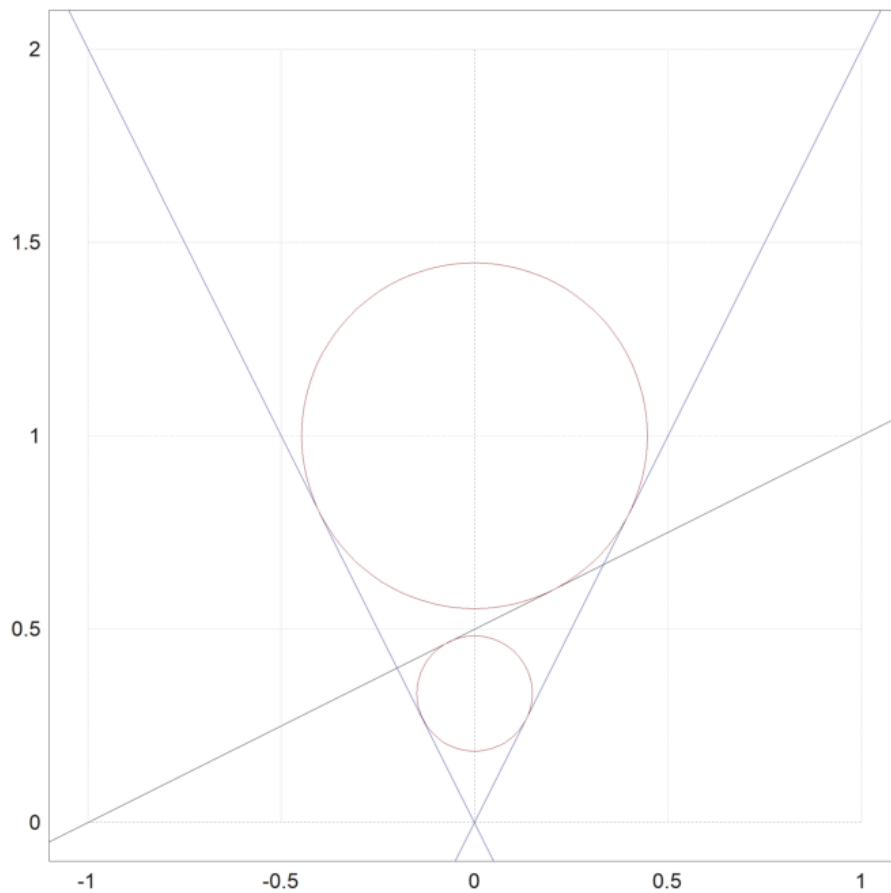
```
[0.333333, 1]
```

```
>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 with Povray

Selanjutnya kita plot 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 adegan dengan tepat.

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

Selanjutnya kita menulis kedua bola tersebut 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 yang dibatasi 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 buat 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)));
```

Lalu kita buat dua titik di mana bola menyentuh bidang. Ini adalah fokus 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 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.

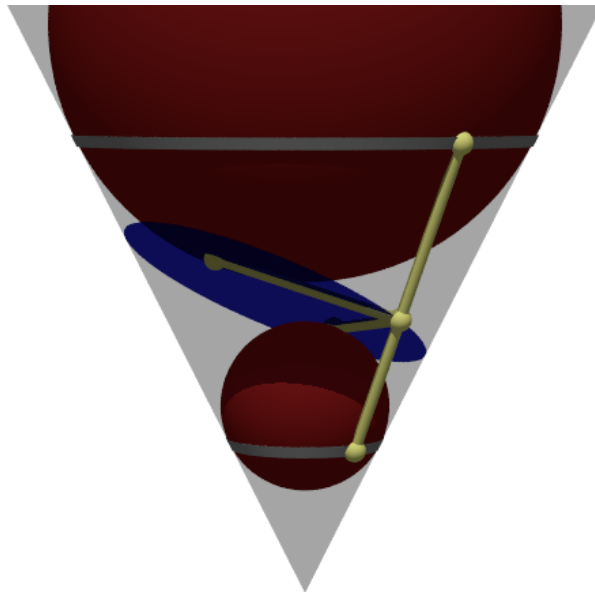
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan pita 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.

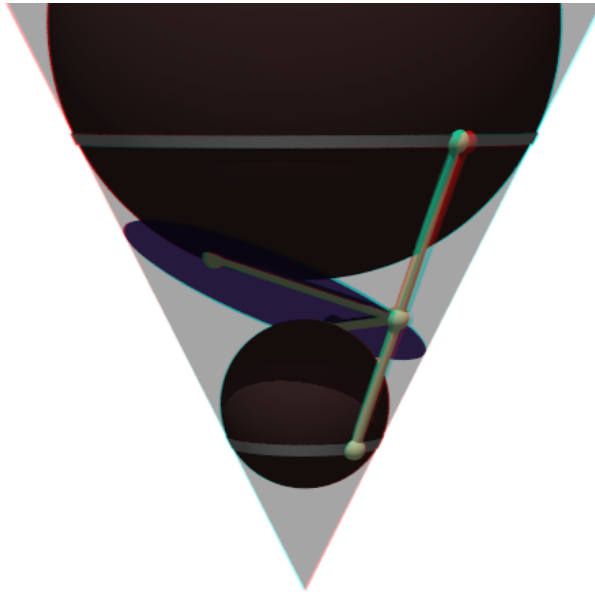
```
>function scene () ...
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g();
pc=povcone([0,0,0],0,[0,0,a],1,"");
vp=[gp[1],0,gp[2]]; dp=gp[3];
writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```



```
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 memerlukan kacamata merah/cyan untuk melihat efek berikut.

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



Contoh 8: Geometri Bumi

Di buku catatan ini, kami ingin melakukan beberapa perhitungan bola. 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, kita 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` (spherical position print).

```
>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)];
>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 bola ke bola ideal lainnya. Vektor ini adalah [heading,distance] 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 bagus. Rutinitas berikut menggunakan perkiraan yang lebih baik. Pada jarak sedekat itu, 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 ...  
^
```

Judulnya ada fungsinya, dengan mempertimbangkan bentuk bumi yang elips. Sekali lagi, kami mencetak dengan cara yang canggih.

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

```
65.34°
```

Sudut suatu segitiga melebihi 180° pada bola.

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

```
180°0'10.77''
```

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

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

Ada fungsi untuk ini, yang menggunakan garis lintang rata-rata dari 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. Vektor berisi arah dan jarak, keduanya dalam radian. Untuk mendapatkan vektor, kita menggunakan svector. Untuk menambahkan vektor ke suatu 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. Hal yang sama terjadi 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 lihat 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 429,66km. Kami mendapatkan perkiraan yang bagus.

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

```
Commands must be separated by semicolon or comma!  
Found: // perkiraan jarak Tugu Jogja - Monas Jakarta (character 32)  
You can disable this in the Options menu.  
Error in:  
esdist(Tugu,Monas)->" km" // perkiraan jarak Tugu Jogja - Monas Jakarta ...  
^
```

Headingnya sama dengan yang dihitung di Google Earth.

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

```
294°17'2.85''
```

Namun kita tidak lagi mendapatkan posisi sasaran yang tepat, jika kita menambahkan heading dan jarak ke posisi semula. Hal ini terjadi karena kita tidak menghitung fungsi invers secara tepat, namun melakukan perkiraan jari-jari bumi di sepanjang lintasan.

```
>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 ingin mengambil jalur terpendek. Bayangkan, Anda terbang NE mulai dari titik mana saja di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti arah yang konstan!

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

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

Sekarang kita tambah 10 kali sepersepuluh jarak, pakai jurusan Monas, kita sampai di Tugu.

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

Hasilnya jauh sekali.

```
>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 yang mempunyai garis lintang yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

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

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

```
79.69°
```

Namun, jika kita mengikuti pembacaan kompas ini, kita akan berputar ke kutub utara! Jadi kita harus menyesuaikan arah perjalanan kita. Untuk tujuan kasarnya, kita sesuaikan 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 arah yang sama terlalu lama.

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

```
0.203km
```

Kita mendapatkan perkiraan yang baik, jika kita menyesuaikan arah 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 sepanjang lingkaran besar menuju Monas dengan fungsi navigasi.

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

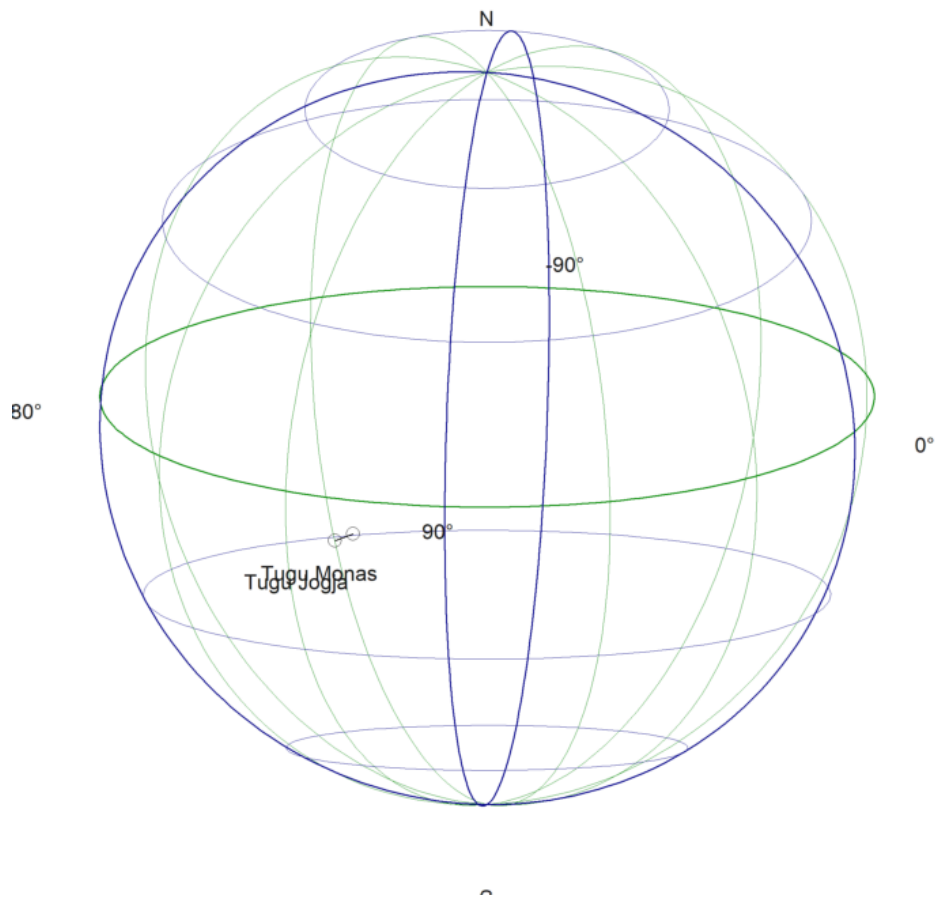
```
S 7°46.998' E 110°21.966'  
S 7°37.422' E 110°0.573'  
S 7°27.829' E 109°39.196'  
S 7°18.219' E 109°17.834'  
S 7°8.592' E 108°56.488'  
S 6°58.948' E 108°35.157'  
S 6°49.289' E 108°13.841'  
S 6°39.614' E 107°52.539'  
S 6°29.924' E 107°31.251'  
S 6°20.219' E 107°9.977'  
S 6°10.500' E 106°48.717'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

Kita 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
```

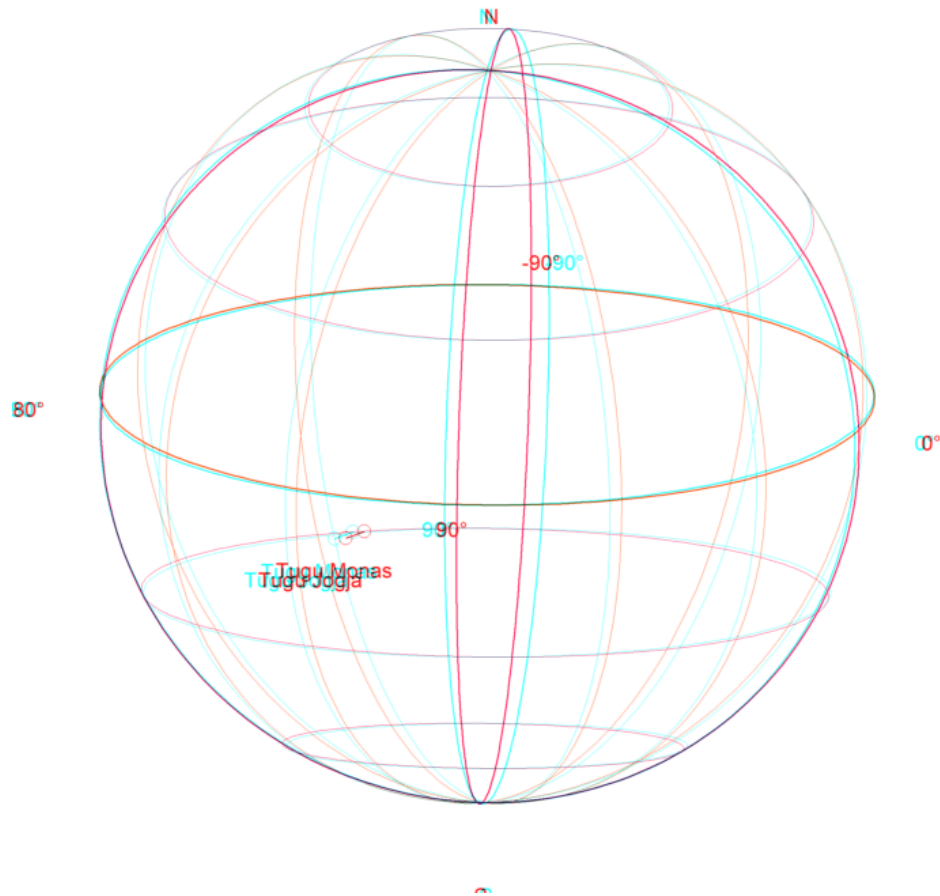
Sekarang plot semuanya.

```
>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 .

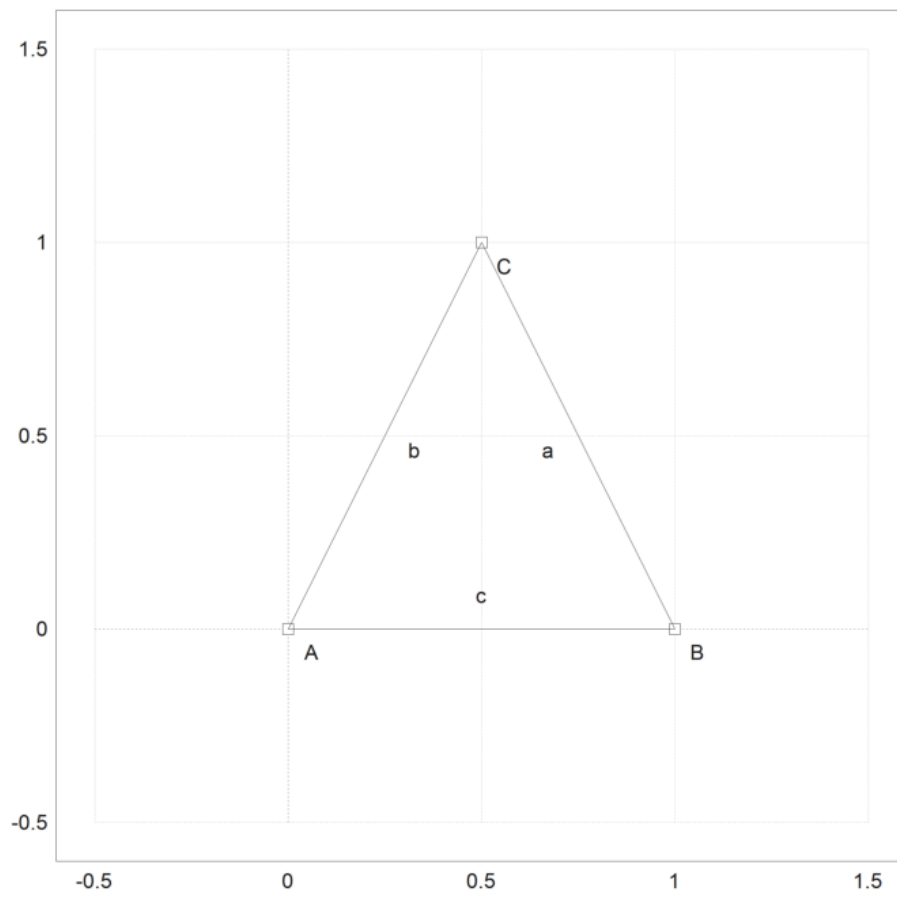
Petunjuk:

- 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.

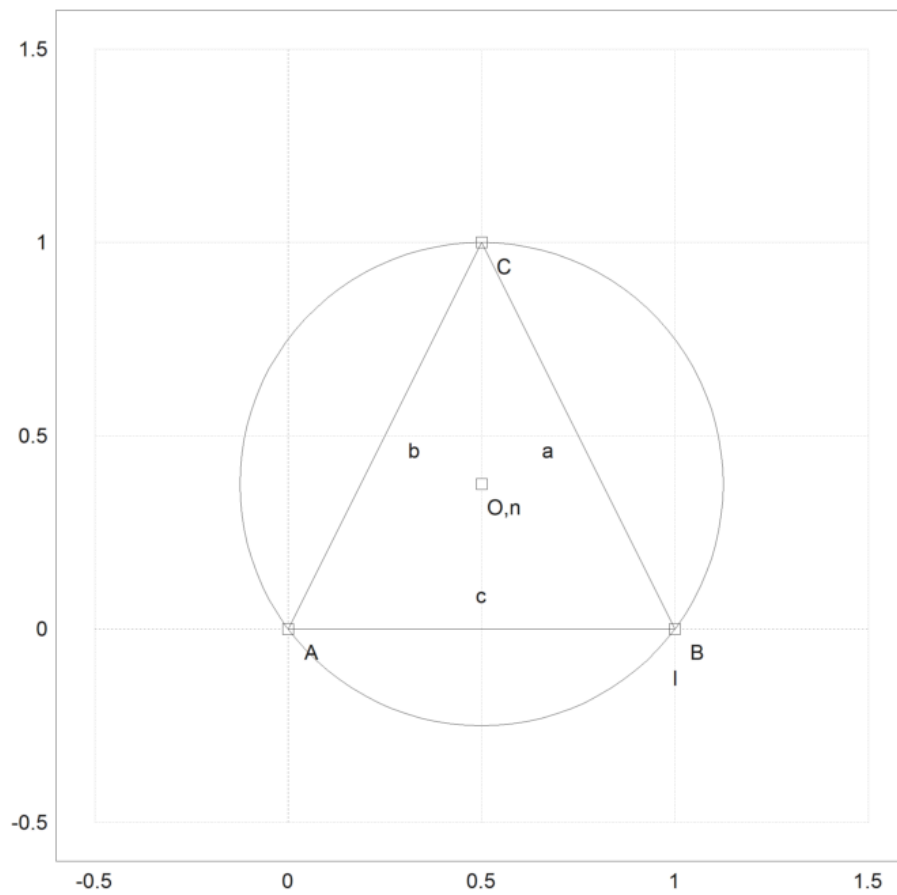
```
>load geometry
```

Numerical and symbolic geometry.

```
>setPlotRange(-0.5,1.5,-0.5,1.5);
>A=[0,0]; plotPoint(A,"A");
>B=[1,0]; plotPoint(B,"B");
>C=[0.5,1]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(C,A,"b")
>aspect(1):
```



```
>l=circleThrough(A,B,C);  
>R=getCircleRadius(l);  
>O=getCircleCenter(l);  
>plotPoint(O,"O,n");  
>plotCircle(l):
```



```
>distance(O,A)
```

```
0.625
```

```
>distance(O,B)
```

```
0.625
```

```
>distance(O,C)
```

```
0.625
```

Terbukti titik pusat lingkaran ke titik sudut jaraknya (r) sama.

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

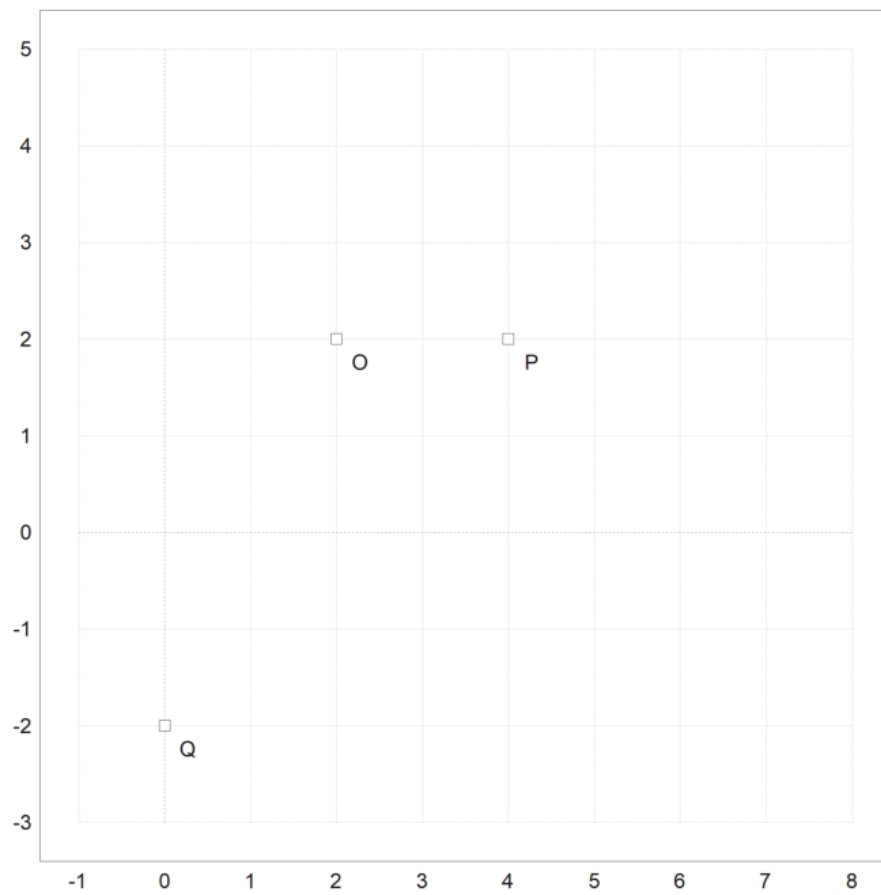
Petunjuk:

- 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
```

```
Numerical and symbolic geometry.
```

```
>setPlotRange(-1,8,-3,5); O=[2,2]; P=[4,2]; Q=[0,-2];
>plotPoint(O,"O"); plotPoint(P,"P"); plotPoint(Q,"Q");
```



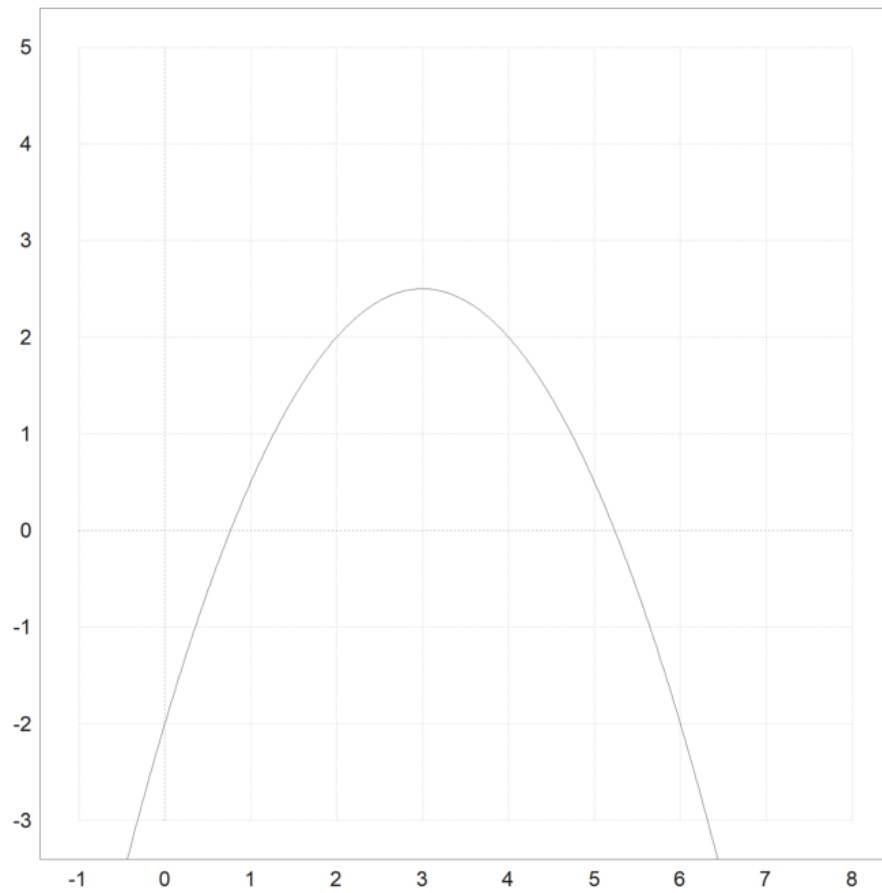
```
>sol &= solve([a*2^2+b*2+c=2,c=-2,a*4^2+b*4+c=2],[a,b,c])
```

```
[[a = - 1/2, b = 3, c = - 2]]
```

```
>function y &=-1/2*x^2+3*x-2
```

$$-\frac{x^2}{2} + 3x - 2$$

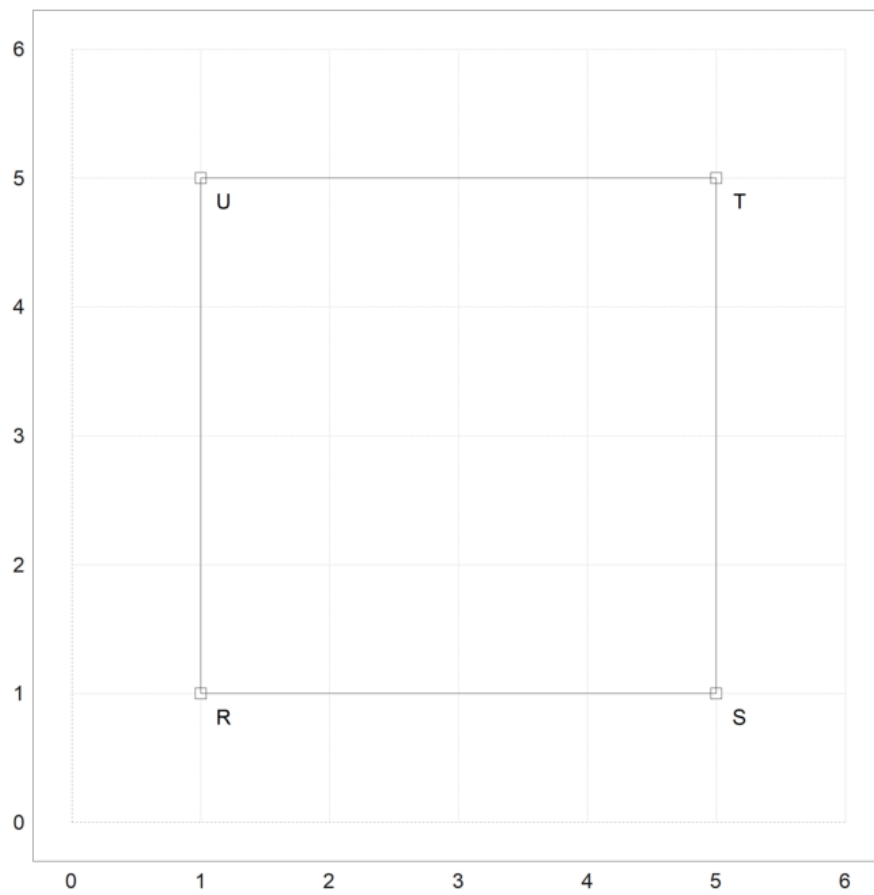
```
>plot2d("-1/2*x^2+3*x-2",-1,8,-3,5):
```

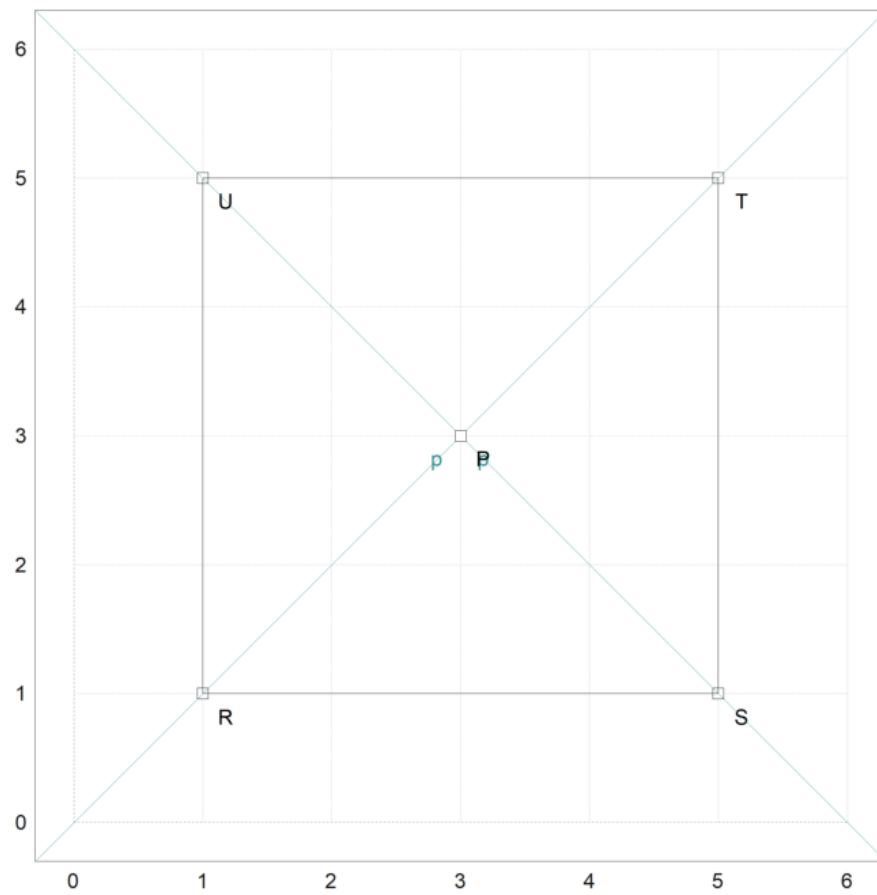
3. Gambarkanlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.

- Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung (sisinya-sisinya 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.

```
>setPlotRange(0,6,0,6);
>R=[1,1]; plotPoint(R,"R");
>S=[5,1]; plotPoint(S,"S");
>T=[5,5]; plotPoint(T,"T");
>U=[1,5]; plotPoint(U,"U");
>plotSegment(R,S,"");
>plotSegment(S,T,"");
>plotSegment(T,U,"");
>plotSegment(U,R,"");
>aspect(1):
```

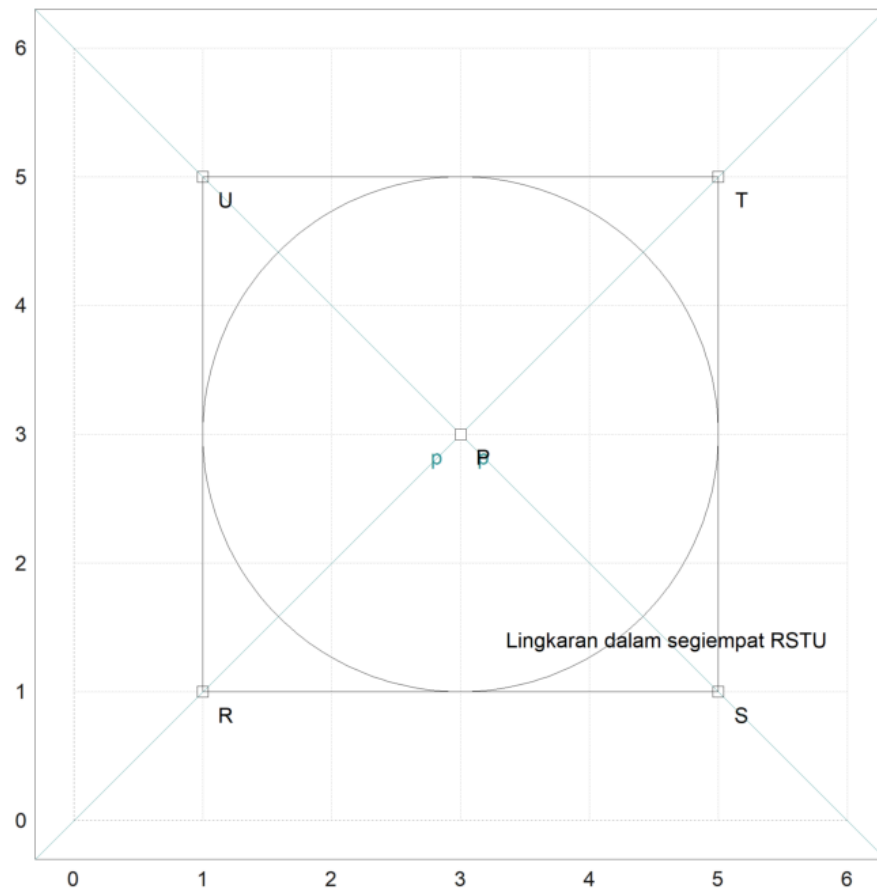


```
>v=angleBisector(R,S,T);  
>w=angleBisector(S,T,U);  
>P=lineIntersection(v,w);  
>color(5); plotLine(v); plotLine(w); color(1);  
>plotPoint(P,"P");
```



Karena keempat garis bagi sudutnya bertemu di satu titik maka segi-4 tersebut adalah segi-4 garis singgung.

```
>r=norm(P-projectToLine(P,lineThrough(R,S)));
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segiempat RSTU");
```



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

```
>RS=norm(R-S) //Panjang sisi RS
```

4

```
>ST=norm(S-T) //Panjang sisi RS
```

4

```
>TU=norm(T-U) //Panjang sisi RS
```

4

```
>UR=norm(U-R) //Panjang sisi RS
```

4

```
>RS.TU
```

16

```
>UR.ST
```

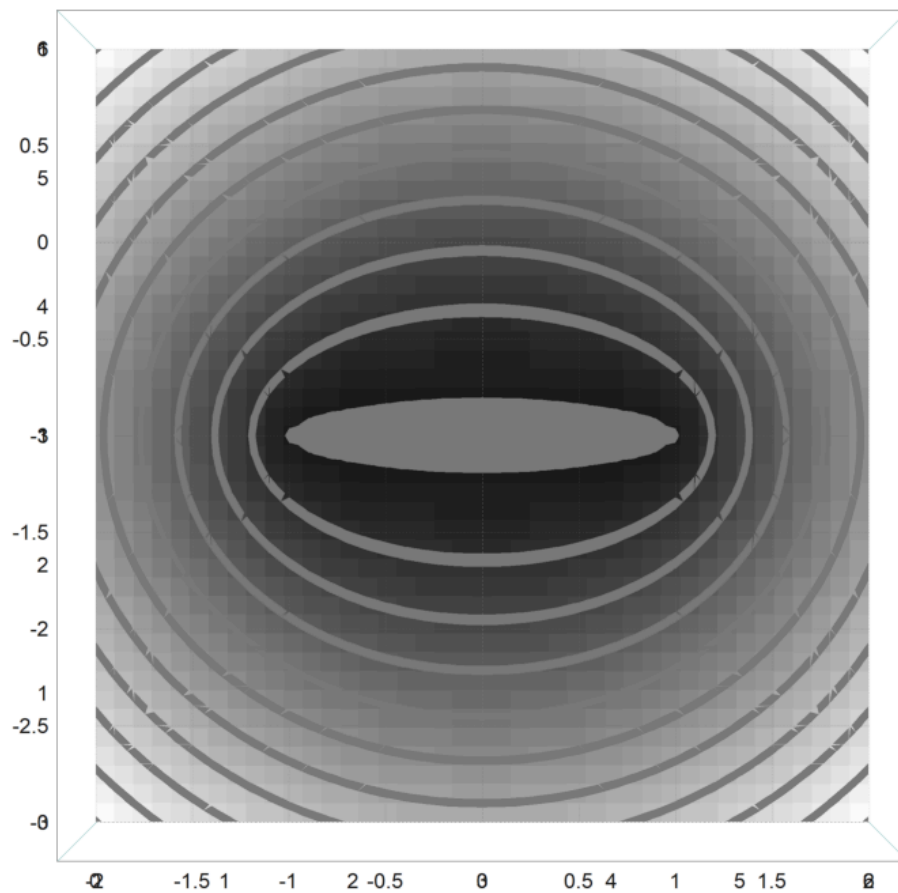
16

Terbukti bahwa hasil kali panjang sisi-sisi yang berhadapan sama, yakni 16

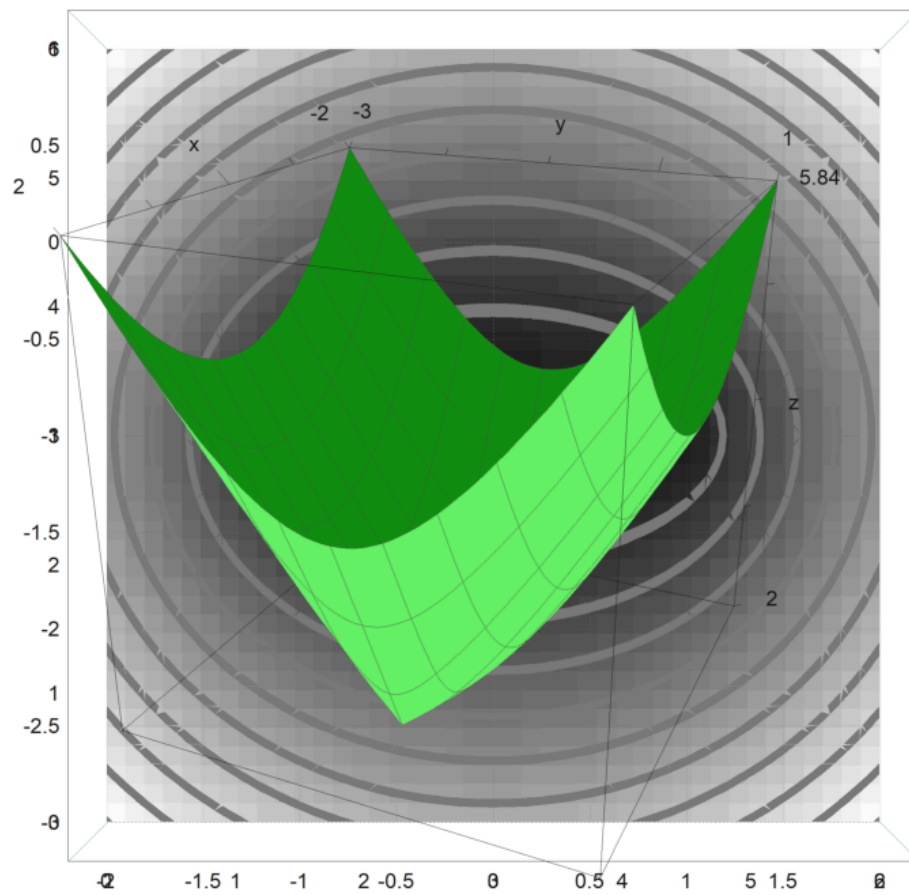
4. Gambarkanlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).

```
>P=[-1,-1]; Q=[1,-1];
>function dl(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):
```

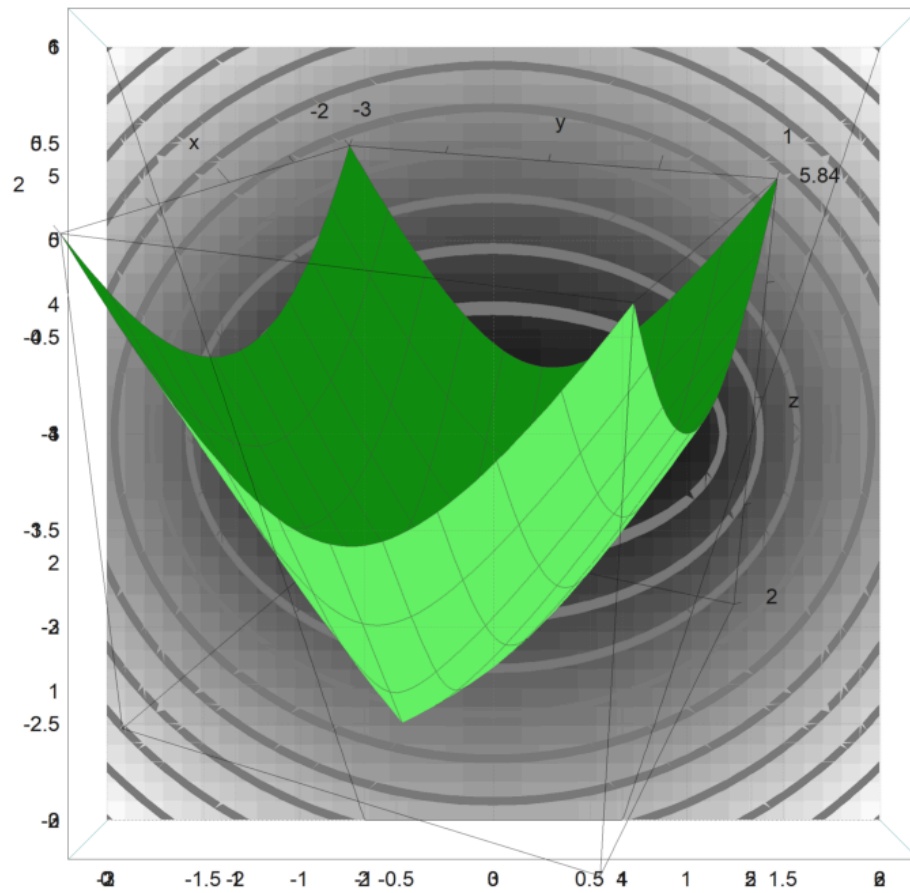


```
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```



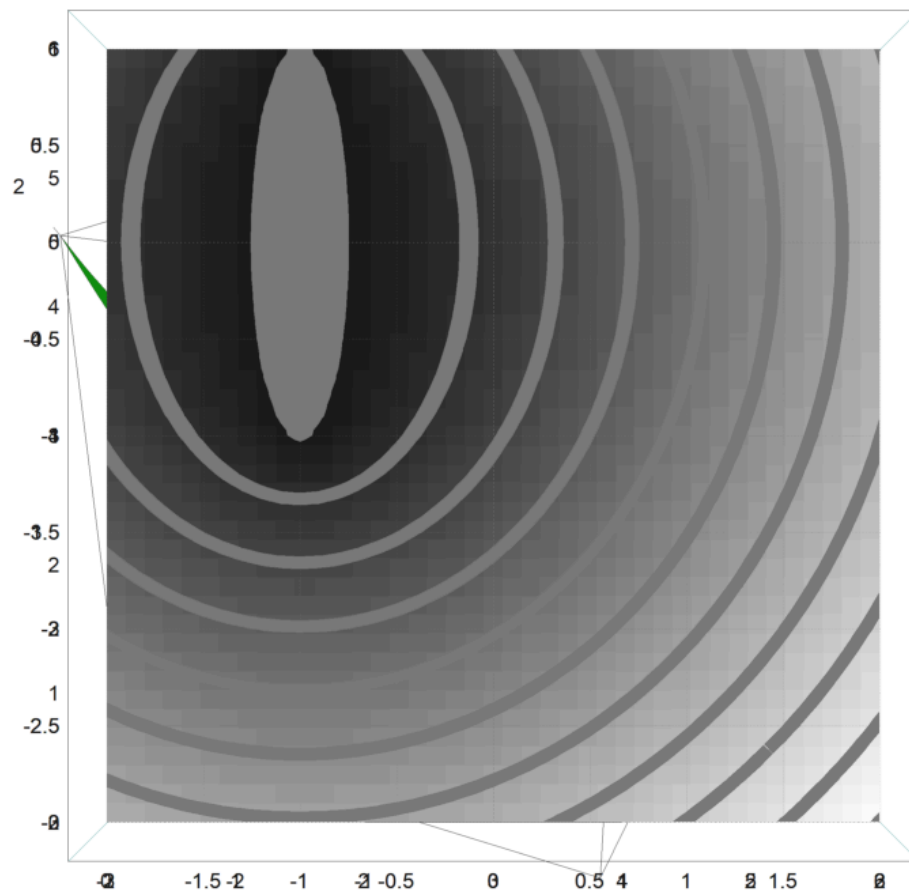
Batasan ke baris PQ

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

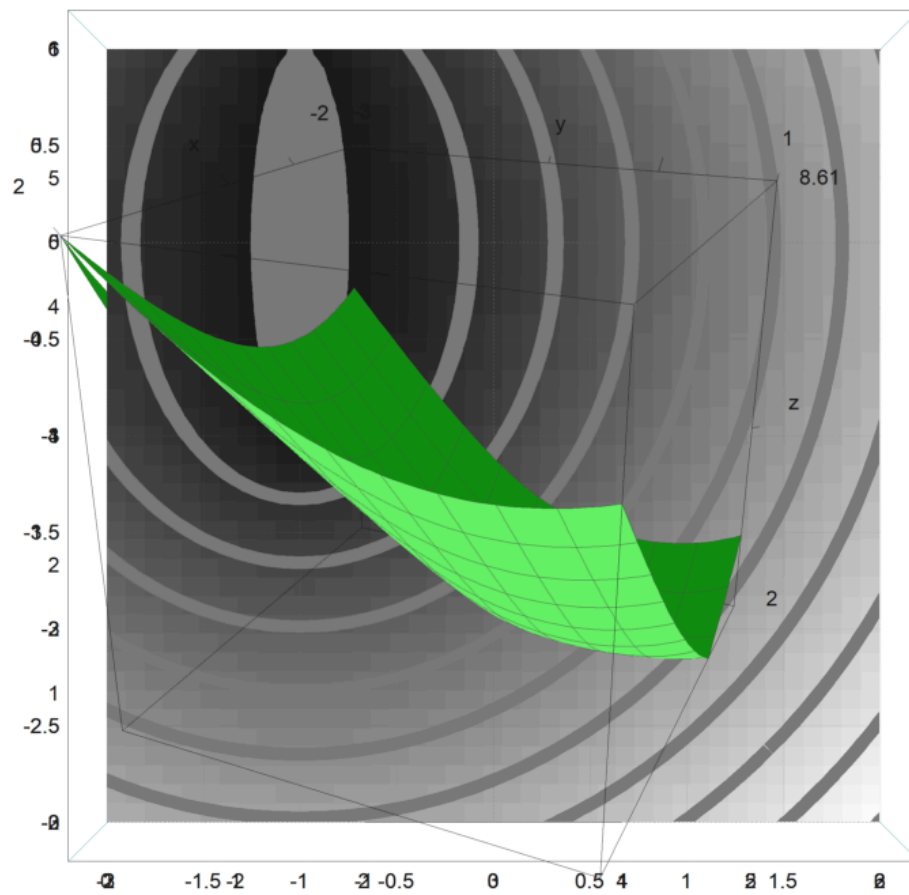


5. Gambarkan suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

```
>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):
```

