UMMI NURROHMAH

24030130095

Pendidikan Matematika C

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 geometeri 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 vektorv 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 lingkran 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 memebntuk 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, tentukan tiga titik dan plot titik-titik tersebut.

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

Kemudian tiga segmen.

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

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

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

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

maksudnya, garis yang melalui B dan C adalah persamaan garis -x+2y=2

Hitung garis tegak lurus melalui A pada BC.

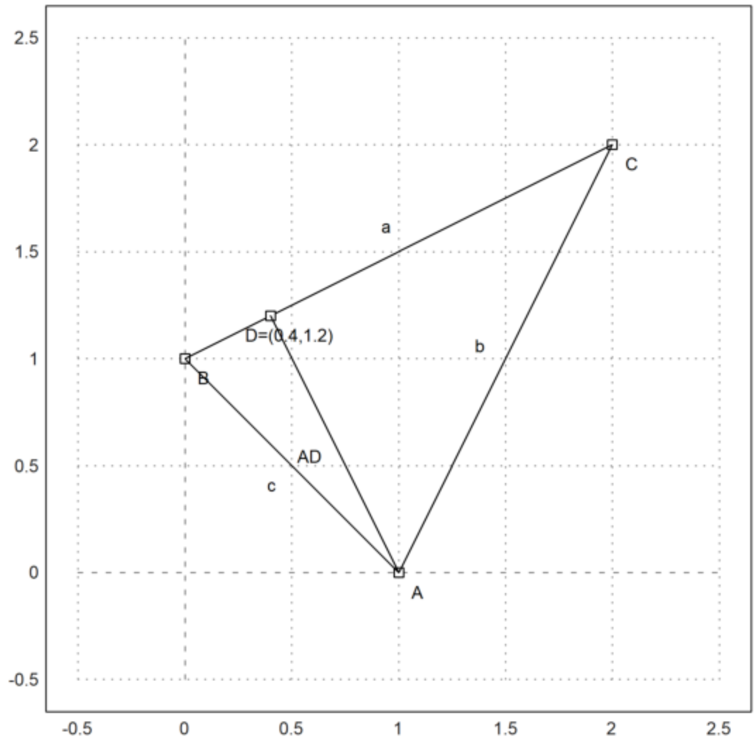
```
>h=perpendicular(A,lineThrough(B,C)); // mendefinisikan garis h tegak lurus BC melalui A
```

Dan titik potongnya dengan BC.

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

Plotkan itu.

```
>plotPoint(D,value=1); // koordinat D ditampilkan
>aspect(1); plotSegment(A,D): // membuat segmen AD lalu menampilkan semua gambar hasil plot...()
```



Hitung luas ABC:

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

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

```
1.5
```

Bandingkan dengan rumus determinan.

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

```
1.5
```

Cara lain menghitung luas segitigas ABC:

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

```
1.5
```

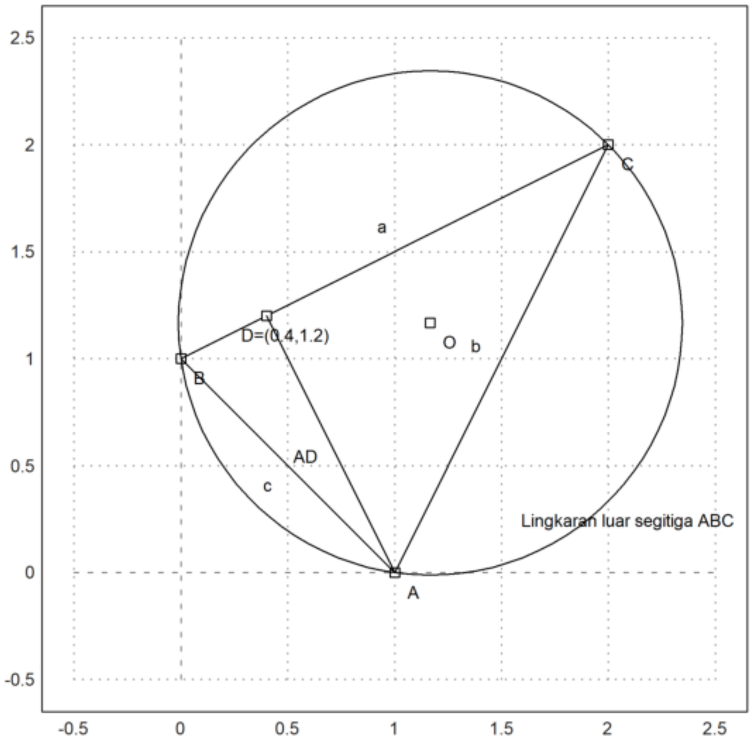
Untuk mengetahui besar sudut C :

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

```
36°52'11.63''
```

Sekarang lingkaran luar dari 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
```

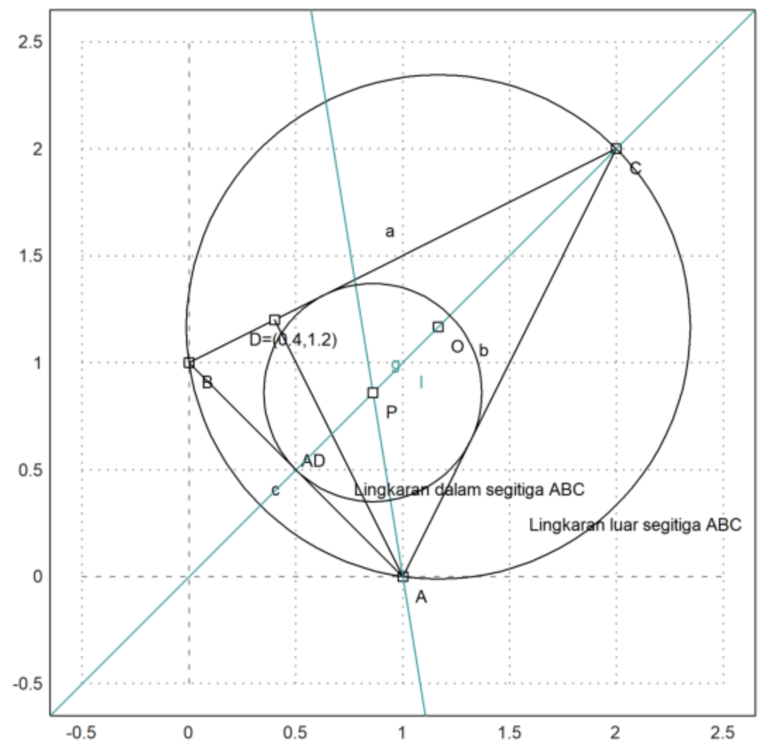
```
[0.86038,  0.86038]
```

Tambahkan apapun ke plot.

```
>color(5); plotLine(l, "l"); plotLine(g, "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
```



1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.
- Titik singgung lingkaran dalam dengan garis a (garis BC) :

```
>d=circleWithCenter(P,r); ...
>a=lineThrough(B,C); ...
>Q=lineCircleIntersections(a,d)// bentuk fungsi titik potong garis-lingkaran jika keduanya sudah did
```

```
[0.632456,  1.31623]
```

- Titik singgung lingkaran dalam dengan garis b (garis AC) :

```
>R=lineCircleIntersections(lineThrough(A,C),d)// misal garis b belum didefinisikan sebelumnya
```

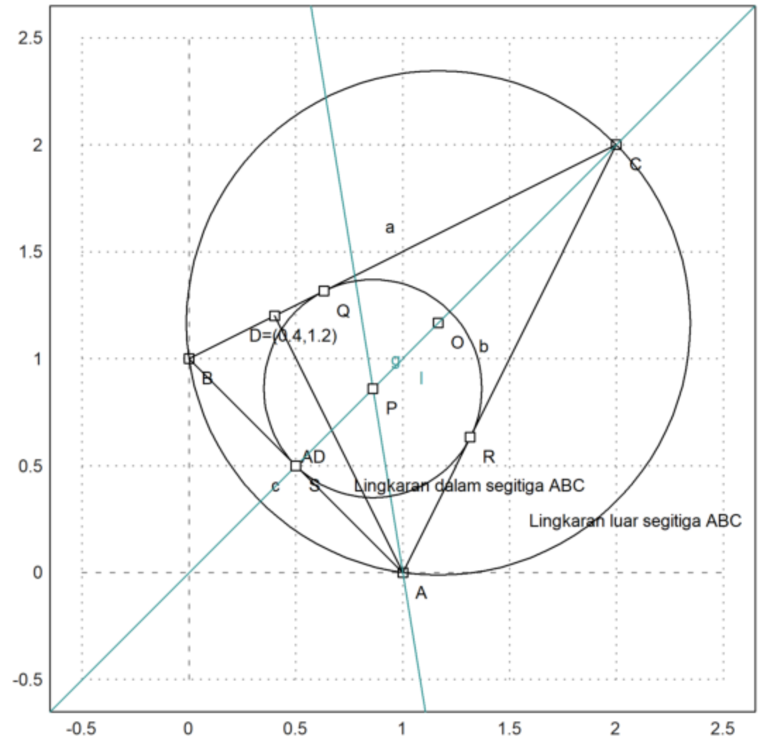
```
[1.31623,  0.632456]
```

- Titik singgung lingkaran dalam dengan garis c (garis AB) :

```
>S=lineCircleIntersections(lineThrough(A,B),circleWithCenter(P,r))// misal garis c dan lingkaran d b
```

```
[0.5,  0.5]
```

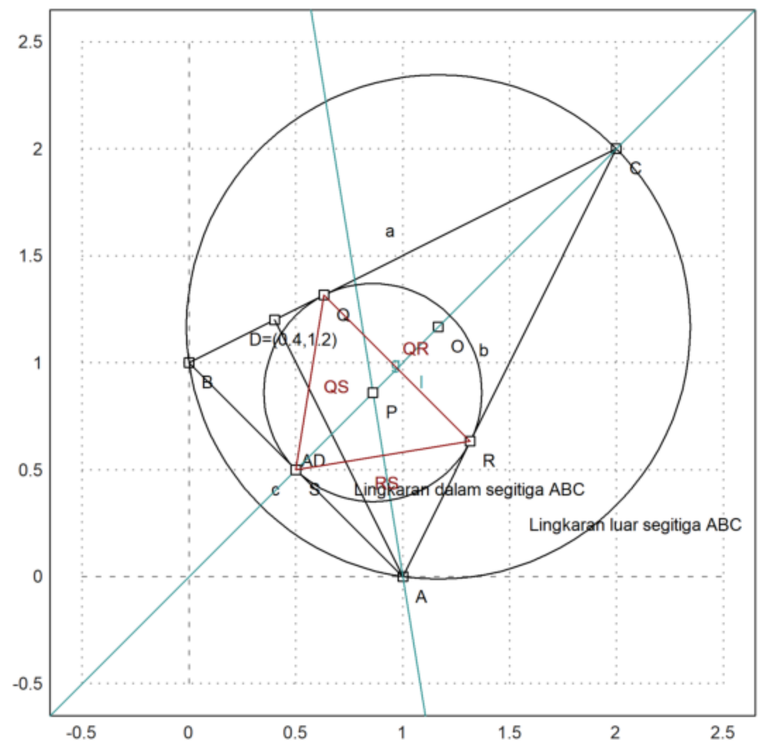
```
>plotPoint(Q,"Q");...
>plotPoint(R,"R");...
>plotPoint(S,"S"):
```



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

Berikut ini adalah gambar segitiga QRS (berwarna merah) :

```
>plotSegment(Q,R); plotSegment(Q,S); plotSegment(R,S); color(3):
```



Untuk mengetahui jenis segitiganya, kita bisa identifikasi dari ukuran sudut dan atau panjang sisi segitiganya.
- Besar sudut Q,R dan S berturut-turut adalah :

```
>degprint(computeAngle(R,Q,S)), degprint(computeAngle(Q,R,S)), degprint(computeAngle(Q,S,R))
```

```
54°13’2.91’’
54°13’2.91’’
71°33’54.18’’
```

Berdasarkan besar sudutnya, segitiga QRS merupakan segitiga lancip, sekaligus segitiga sama kaki dengan besar sudut Q=R.

- Panjang sisi QR, RS, QS berturut-turut adalah :

```
>distance(Q,R), distance(R,S), distance(Q,S)
```

```
0.966999966873
0.826905214631
0.826905214631
```

Berdasarkan panjang sisinya, segitiga QRS merupakan segitiga sama kaki dengan panjang sisi RS=QS.

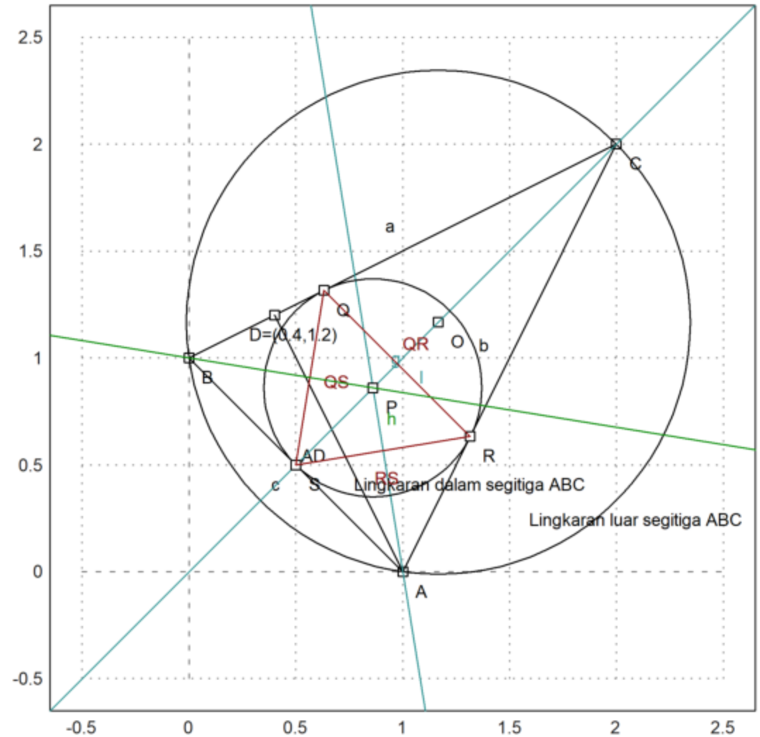
3. Hitung luas segitiga tersebut.

```
>areaTriangle(Q,R,S)//menghitung luas segitiga dengan fungsi geometri analitik untuk luas
```

```
0.324341649025
```

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

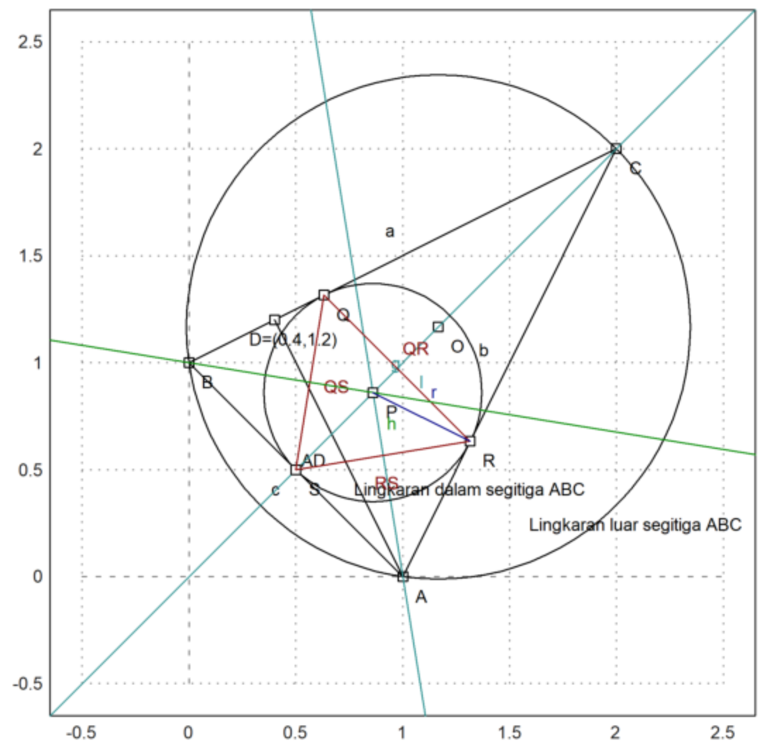
```
>h=angleBisector(A,B,C); // garis bagi <ABC
>plotLine(h, "h"); color(4):
```



Garis bagi sudut yang ke tiga adalah garis bagi sudut ABC (garis berwarna hijau). Dapat dilihat bahwa garis bagi sudut ketiga juga melalui titik pusat lingkaran dalam (titik P).
Jadi dapat disimpulkan bahwa perpotongan garis bagi sudut segitiga merupakan titik pusat lingkaran dalam segitiga tersebut.

5. Gambar jari-jari lingkaran dalam.

```
>plotSegment(P,R,"r"); color(5):
```



Jari-jari lingkaran dalamnnya saya beri label r (garis berwarna biru), yang juga merupakan segmen garis PR (bisa juga segmen gaaris PQ atau PS)

6. Hitung luas lingkaran luar dan luas lingkaran dalam segitiga ABC. Adakah hubungan antara luas kedua lingkaran tersebut dengan luas segitiga ABC?
- Luas Lingkaran Luar Segitiga ABC :

$$L_1 = \pi R^2$$

dengan R=AO=BO=CO

```
>pi*norm(A-O)^2
```

```
4.36332312999
```

- Luas Lingkaran Dalam Segitiga ABC :

$$L_1 = \pi r^2$$

dengan r=PQ=PR=PS

```
>pi*norm(P-Q)^2
```

```
0.81601903655
```

- Luas Segitiga ABC :

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

```
1.5
```

Hubungan antara ketiganya:
Luas lingkaran luar segitiga > Luas segituga > luas lingkaran dalam segitiga

Contoh 2: Geometri Simbolik

Kita dapat menghitung geometri eksak dan sibolik menggunakan Maxima.

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

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

Fungsi untuk garis dan lingkaran bekerja sama seperti fungsi Euler, tetapi menyediakan perhitungan simbolik

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

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

Kita dapat dengan mudah mendapatkan persamaan untuk sebuah garis.

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

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

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

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

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

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

```
>$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
```

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

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

```
 2  6
[-, -]
 5  5
```

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

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

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

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

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

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

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

$$1.178511301977579$$

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

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

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

$$y=x$$

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

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

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

```
[0.86038,  0.86038]
```

Garis dan Lingkaran yang Berpotongan.

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

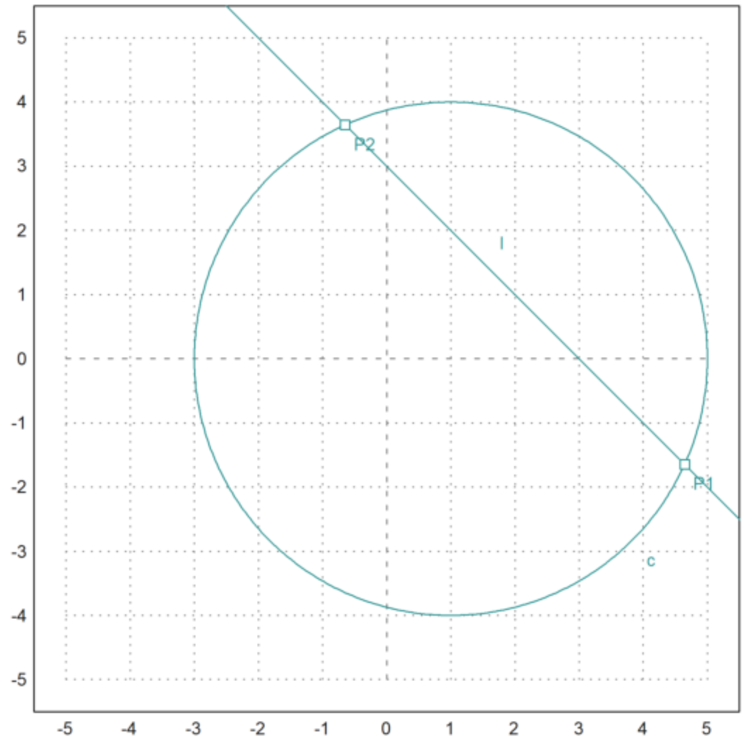
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l, "l");
```

Pertemuan antara sebuah 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.

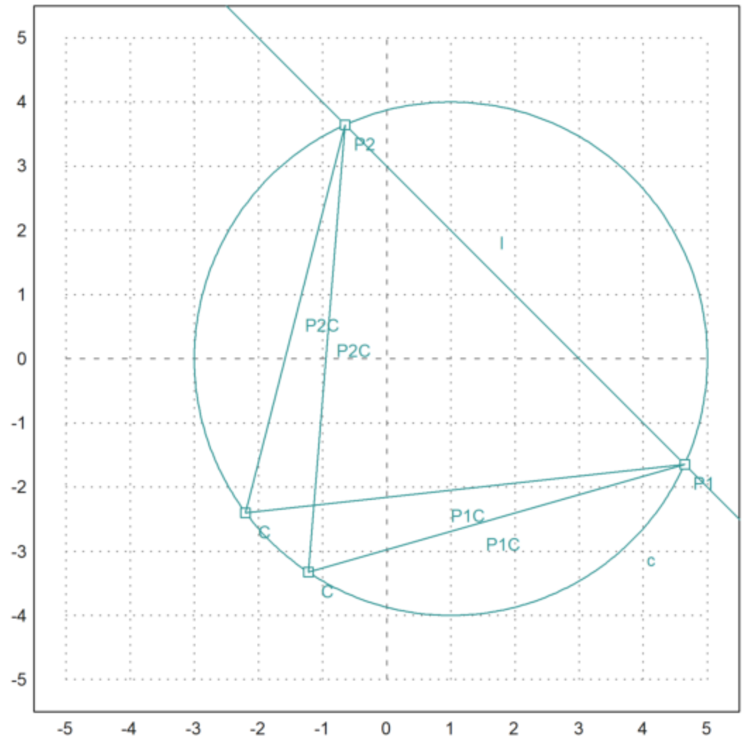
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
>degprint(computeAngle(P1,C,P2))
```

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;
```

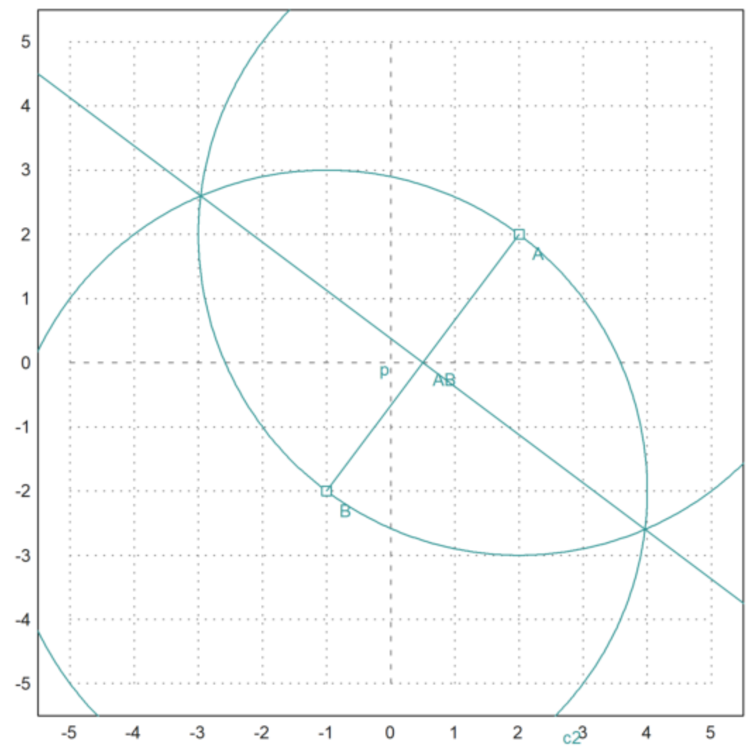


Garis Sumbu

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 melallui 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 lakukan hal yang sama di Maxima dengan koordinat umum.

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

Persamaan untuk titik-titik potongan cukup rumit. Namun, kita bisa menyederhanakannya jika kita menyelesaikan untuk y.

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

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

Ini memang sama dengan tegak lurus tengah, yang dihitung dengan cara yang sepenuhnya berbeda.

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

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

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

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

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:

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

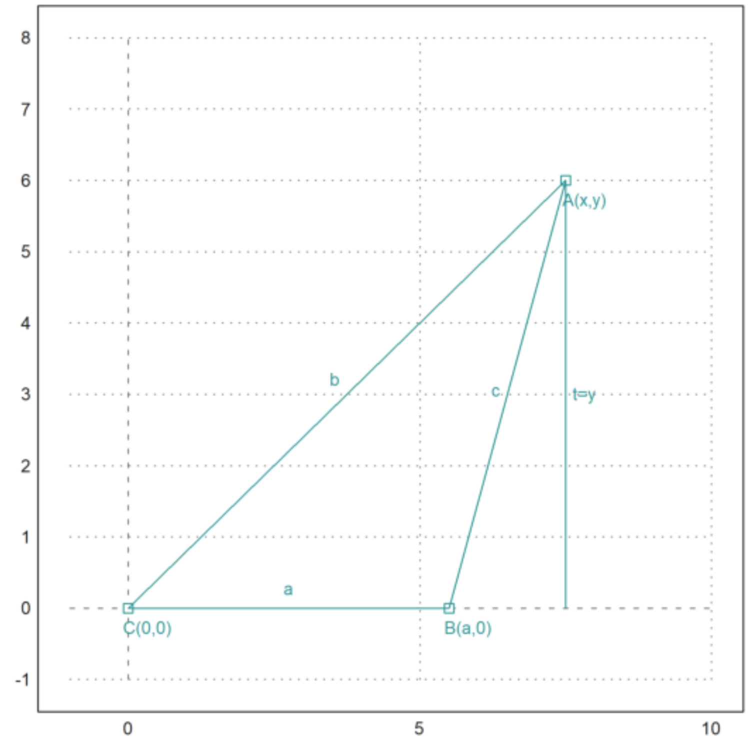
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

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])
```

```
             2    2    2
          - c  + b  + a
[[x = --------------, y =
```

```
                     2 a
          4       2  2       2  2    4       2  2    4
    sqrt(- c  + 2 b  c  + 2 a  c  - b  + 2 a  b  - a )
  - -------------------------------------------------],
                           2 a
          2    2    2
       - c  + b  + a
  [x = --------------, y = 
            2 a
        4       2  2       2  2    4       2  2    4
  sqrt(- c  + 2 b  c  + 2 a  c  - b  + 2 a  b  - a )
  -------------------------------------------------]]
                         2 a
```

Ekstrak solusi y.

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

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

Kita mendapatkan rumus Heron.

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

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

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

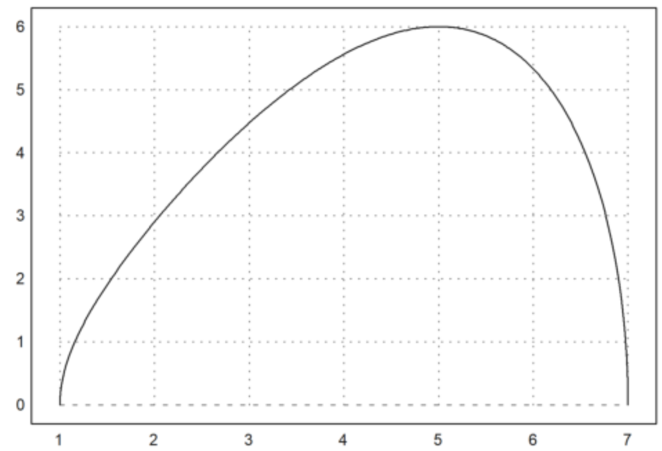
$$Luas=\frac{3\,\sqrt{39}}{4}$$

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

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

Dan juga jelas bahwa ini adalah segitiga dengan luas maksimum 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 (1<= x <=7)
```



Kasus umum juga berlaku.

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

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

Sekarang mari kita temukan himpunan semua titik dimana b+c=d untuk beberapa konstanta d. Sudah diketahui bahwa ini adalah sebuah elips.

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

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

Dan buat fungsi dari ini.

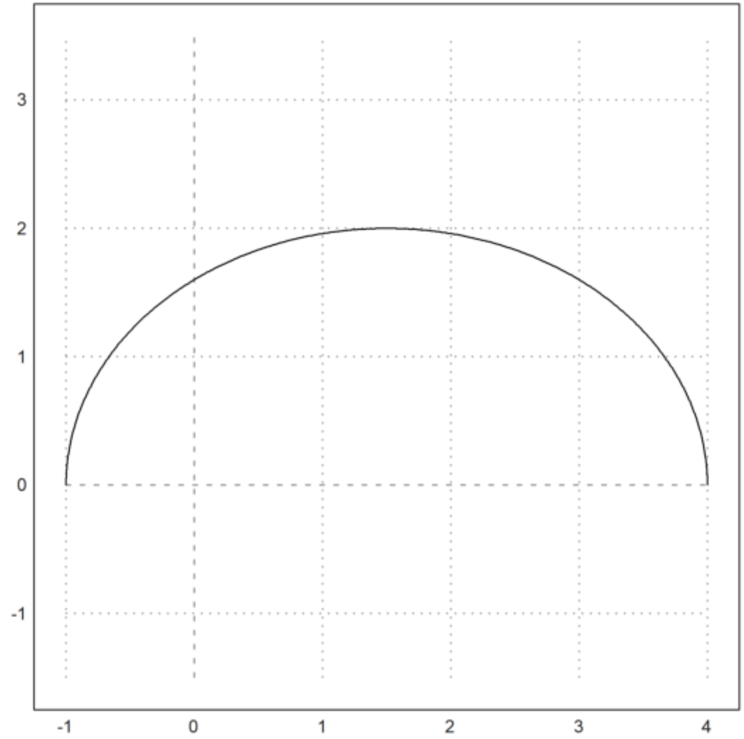
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

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

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

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

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



Kita bisa memeriksa persamaan umum untuk elips ini, yaitu.

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

dimana (xm,ym) adalah pusatnya, dan u dan v dalah sumbu setengah.

```
>$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 dengan demikian luas segitiga, adalah maksimal untuk x=0. Jadi, luas segitiga dengan a+b+c=d adalah maksimal jika segitiga tersebut sama kaki. Kita ingin menurunkannya secara analitis.

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

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

Kita mendapatkan beberapa titik minimum, yang terkait dengan segitiga dengan salah 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, yang memaksimalkan H(a,b,c)^2 dengan mempertimbangkan a+b+d=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])
```

```
                        d      d
          [[a = 0, b = -, c = -, la = 0],
                        2      2
     d              d                 d      d
  [a = -, b = 0, c = -, la = 0], [a = -, b = -, c = 0, la = 0],
     2              2                 2      2
                                3
     d      d      d           d
  [a = -, b = -, c = -, la = ---]]
     3      3      3          108
```

Kita dapat membuat plot dari situasi tersebut.

Pertama, tentukan titik-titik dalam Maxima.

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

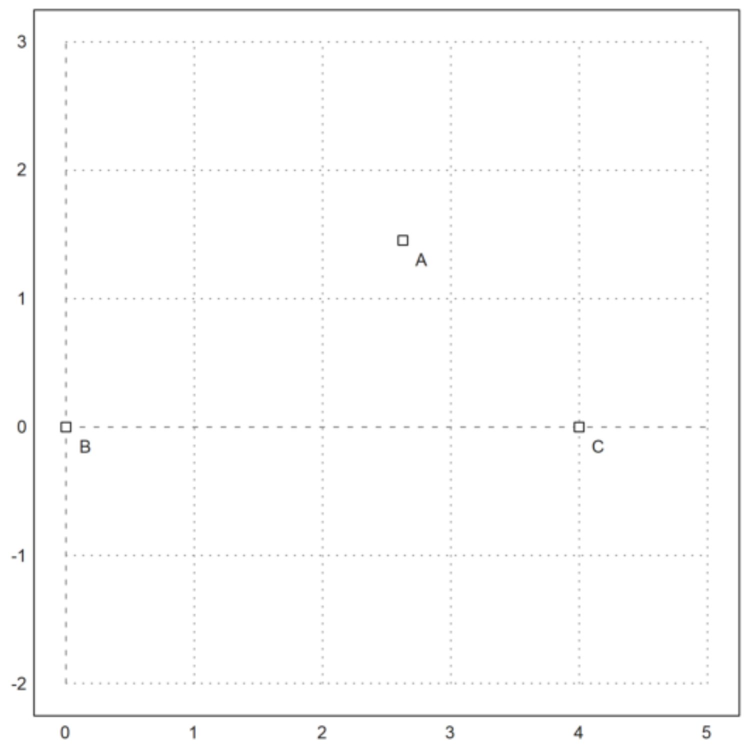
$$\left[\frac{-c^2+b^2+a^2}{2\,a} , \frac{\sqrt{-c^4+2\,b^2\,c^2+2\,a^2\,c^2-b^4+2\,a^2\,b^2-a^4}}{2\,a}\right]$$

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

$$[a,0]$$

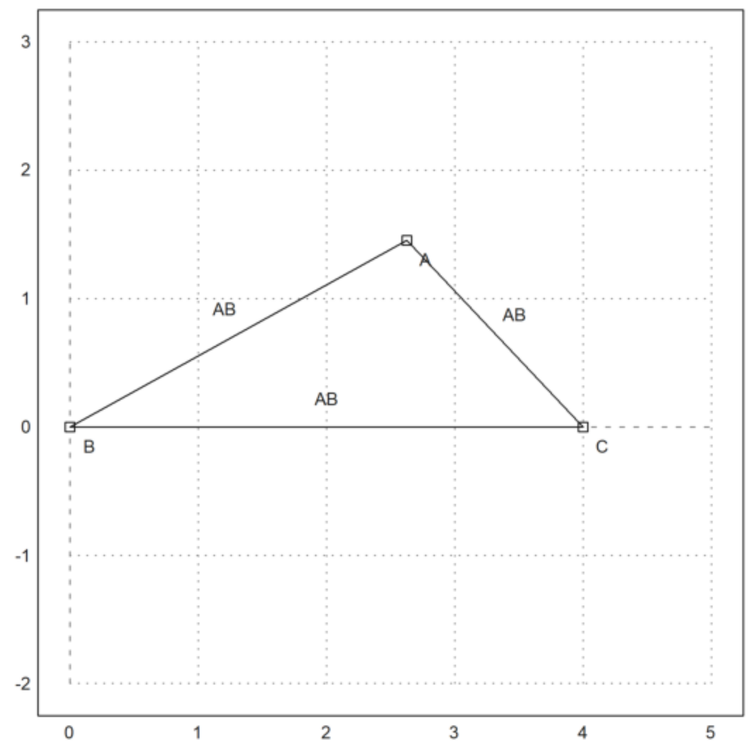
Selanjutnya, tetapkan rentang plot, dan gambarkan titik-titik tersebut.

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



Plot the segments.

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



Hitung tegak lurus tengah di Maxima.

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

Dan pusat lingkaran.

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

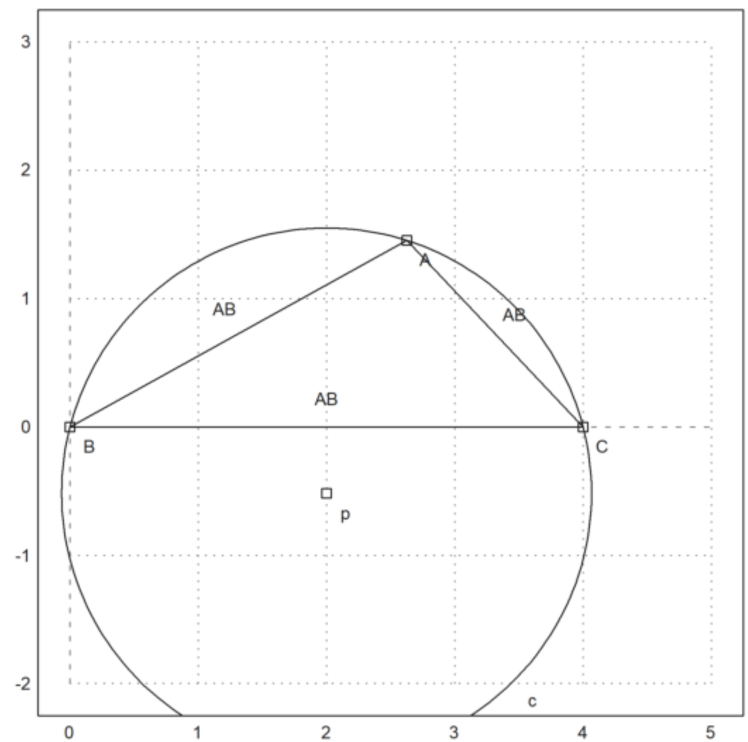
Kita mendapatkan rumus untuk jari-jari lingkaran juring.

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

$$\frac{i\,a\,b\,c}{\sqrt{c-b-a}\,\sqrt{c-b+a}\,\sqrt{c+b-a}\,\sqrt{c+b+a}}$$

Marilah kita tambahkan ini ke dalam plot.

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



Dengan menggunakan geometri, kita menurunkan rumus sederhana.

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

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

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

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

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari sembarang segitiga yang tidak sama kaki. Ini adalah garis pusat segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk ortosenter, pusat juring, centroid, titik Exeter, dan pusat lingkaran sembilan titik segitiga.

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

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

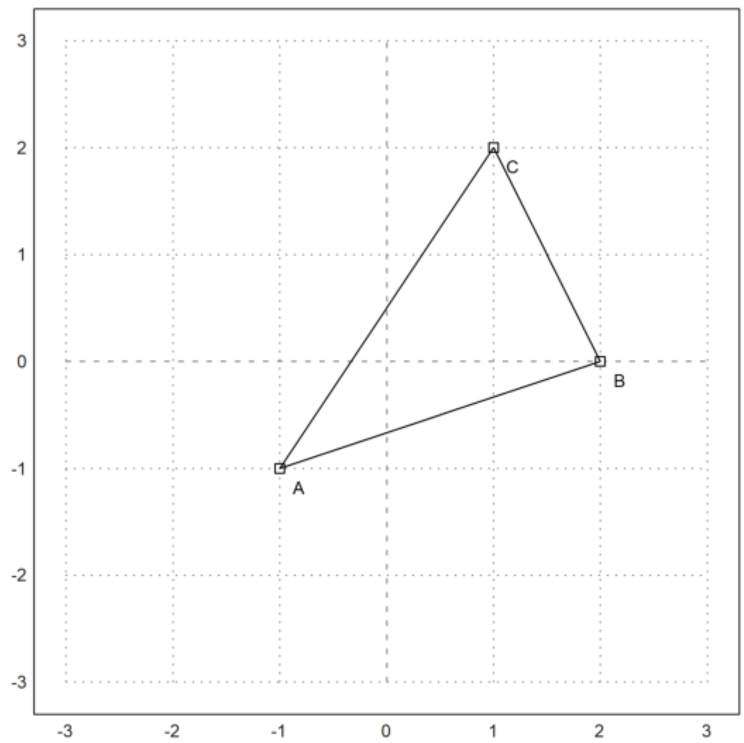
```
>A::=[-1,-1]; B::=[2,0]; C::=[1,2];
```

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

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



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

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

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

Kita dapat menghitung koefisien dari sisi c

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

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

Dan juga mendapatkan rumus untuk garis ini.

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

$$3\,y-x=-2$$

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

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

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

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

Sekarang kita menghitung lingkaran juring dari segitiga ABC.

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

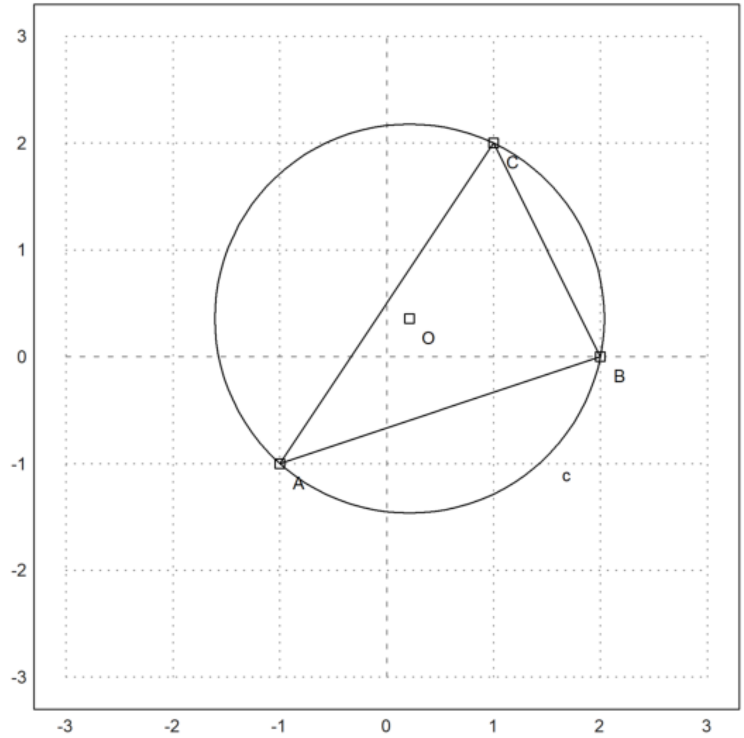
$$\left(y-\frac{5}{14}\right)^2+\left(x-\frac{3}{14}\right)^2=\frac{325}{98}$$

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

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

lot lingkaran dan pusatnya. Cu dan U bersifat simbolik. Kita evaluasi ekspresi ini untuk Euler.

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



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

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

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

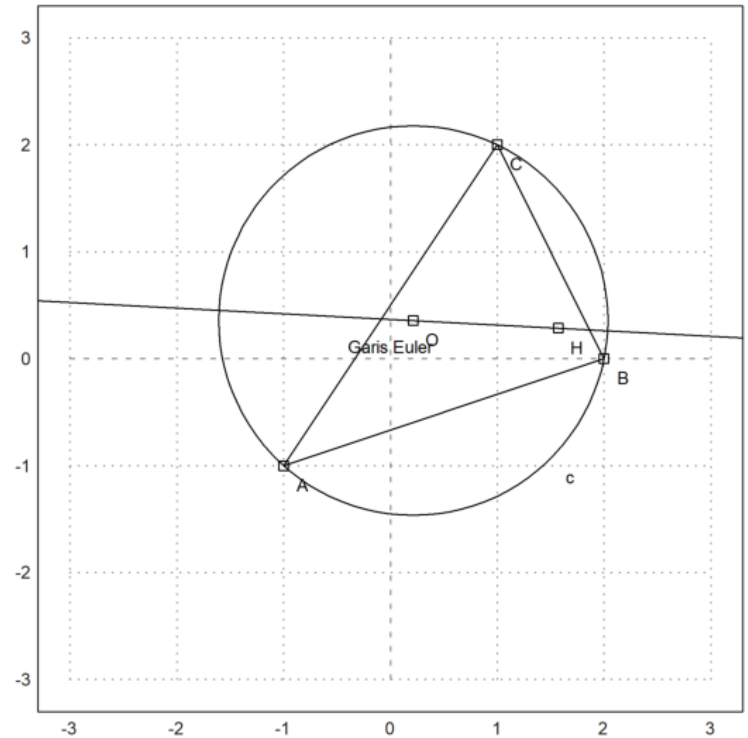
Sekarang kita bisa menghitung garis Euler dari segitiga.

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

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

Tambahkan ini ke plot kita.

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

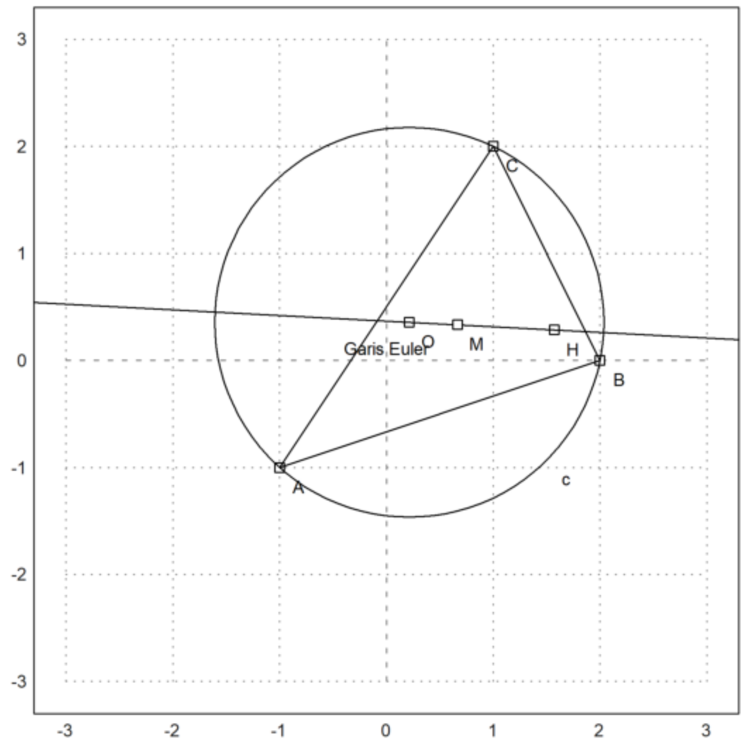


Pusat gravitas harus berada di garis ini.

```
>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 menyatakan bahwa MH=2*MO. Kita perlu menyederhanakan dengan radcan untuk mencapai ini.

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

$$2$$

Fungsi-fungsi tersebut 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 pusat lingkaran dalam tidak terlalu rapi.

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

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

Marilah kita hitung juga ekspresi untuk jari-jari lingkaran yang terinskripsi.

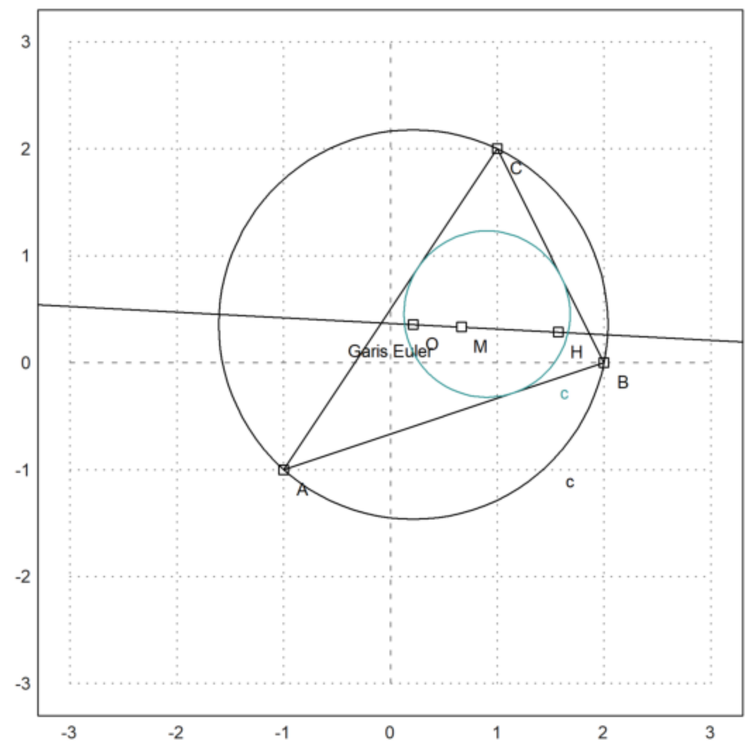
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
```

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

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

Marilah kita tambahkan ini ke dalam plot.

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



Parabola

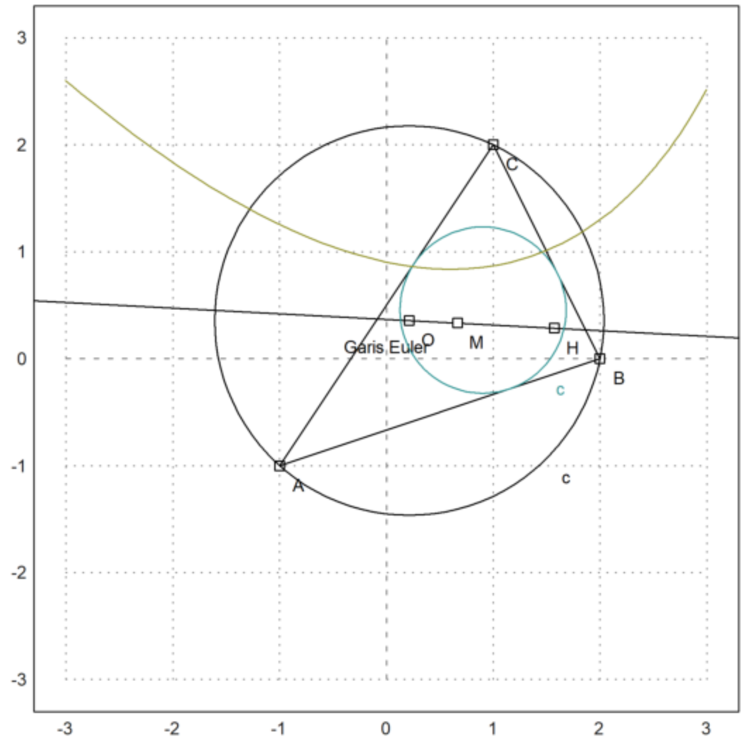
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Ini seharusnya menjadi suatu fungsi, tetapi penyelesai bawaan Maxima hanya dapat menemukan solusi jika kita mengkuadratkan persamaan. Akibatnya, kita mendapatkan solusi palsu.

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

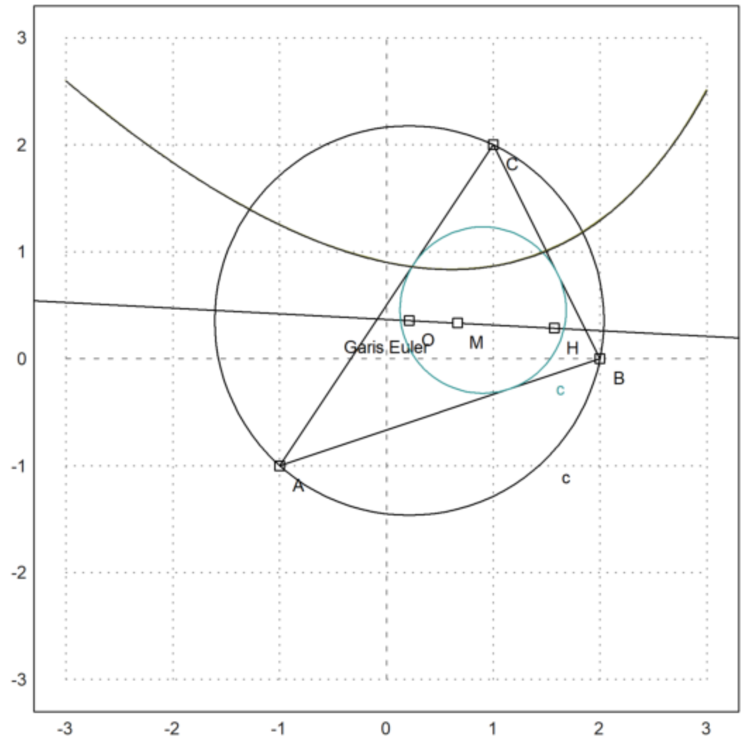
```
[y = - 3 x - sqrt(70) sqrt(9 - 2 x) + 26,
                       y = - 3 x + sqrt(70) sqrt(9 - 2 x) + 26]
```

Solusi pertama adalah

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

Menambahkan solusi pertama ke plot menunjukkan bahwa itu memang jalur yang kita cari. Teori menyatakan bahwa ini 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
```

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

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

$$2.135605779339061$$

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

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

```
>dU2AB &= distance(T,U); $fullratsimp(dU2AB), $float(%) // jatak 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 oleh pembicaraan N.J. Wildberger. Dalam bukunya 'Divine Proportions', Wildberger mengusulkan untuk mengganti konsep klasik tentang jarak dan sudut dengan kuadran dan penyebaran. Menggunakan ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap 'rasional'.

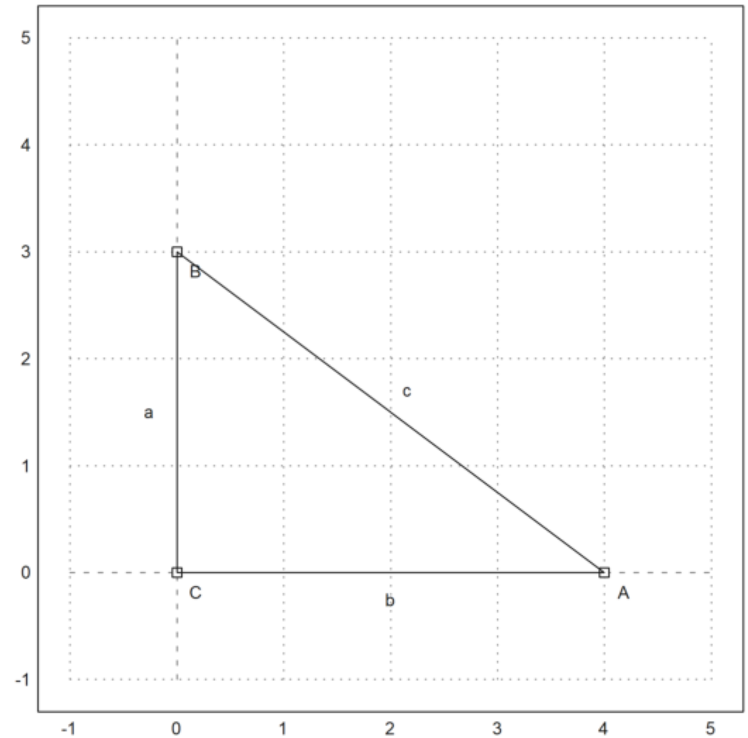
Dalam tulisan berikut, saya akan memperkenalkan konsep-konsep ini dan menyelesaikan beberapa masalah. Saya menggunakan perhitungan simbolik Maxima di sini, yang menyembunyikan keuntungan utama dari trigonometri rasional bahwa perhitungan dapat dilakukan hanya dengan kertas dan pensil. Anda diundang untuk memeriksa hasilnya tanpa komputer.

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

```
>load geometry;
```

Untuk pengenalan pertama, kita menggunakan segitiga siku-siku dengan proporsi Mesir yang terkenal 3, 4 and 5. Perintah berikut adalah perintah Euler untuk memplot geometri bidang yang terdapat dalam file Euler "geometry.e".

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



TEntu saja,

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

di mana wa adalah sudut di A. Cara biasa untuk menghitung sudut ini adalah dengan mengambil invers fungsi sinus. Hasilnya adalah sudut yang sulit dicerna, yang hanya dapat dicetak secara mendekati.

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

```
36°52'11.63''
```

Trigonometri rasional berusaha menghindari ini.

Konsep pertama dari trigonometri rasional adalah kuadran, yang menggantikan jarak. Sebenarnya, ini hanya jarak yang dikuadratkan. Dalam konteks ini, a, b, dan c menunjukkan kuadran dari sisi-sisi segitiga.

Teorema Pythagoras kemudian menjadi a+b=c.

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

```
25 = 25
```

Konsep kedua dari trigonometri rasional adalah penyebaran. Penyebaran mengukur sudut antara garis-garis. Nilainya adalah 0 jika garis-garis tersebut sejajar, dan 1 jika garis-garis tersebut tegak lurus. Ini adalah kuadrat dari sinus sudut antara kedua garis.

Penyebaran dari garis AB dan AC pada gambar di atas didefinisikan sebagai.

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

di mana a dan c adalah kuadran dari sembarang segitiga siku-siku dengan satu sudut di A.

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

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

Ini tentu lebih mudah dihitung daripada sudutnya. Namun, Anda kehilangan sifat bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita bisa mengubah nilai perkiraan untuk sudut wa menjadi penyebaran, dan mencetaknya sebagai pecahan.

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

```
9/25
```

Hukum kosinus dari trigonometri klasik diterjemahkan menjadi 'hukum silang' berikut.

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

Di sini, a, b, dan c adalah kuadran dari sisi-sisi segitiga, dan sa dalah penyebaran di sudut A. Sisi a, seperti biasa, berlawanan dengan sudut A.

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

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

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

Pada kasus kita, diperoleh :

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

$$1024=1024$$

Marilah kita gunakan hukum silang ini untuk menemukan penyebaran di A. Untuk melakukan ini, kita menghasilkan hukum silang untuk kuadran a, b, dan c, dan menyelesaikannya untuk penyebaran yang tidak diketahui sa.

Anda bisa melakukan ini dengan mudah secara manual, tetapi saya menggunakan Maxima. Tentu saja, kita mendapatkan hasil yang sudah kita miliki.

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

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

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

Kita sudah mengetahui ini. Definisi penyebaran adalah kasus khusus dari hukum silang.

Kita juga bisa menyelesaikan ini untuk a, b, dan c secara umum. Hasilnya adalah rumus yang menghitung penyebaran sudut segitiga yang diberikan kuadran dari ketiga sisinya.

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

$$\left[\left[\frac{168\,bb\,x+36\,bb^2+(-72\,aa-84)\ bb+36\,aa^2-84\,aa+49}{36},\frac{168\,bb\,x+36\,bb^2+(-72\,aa-84)\ bb+36\,aa^2-84\,aa+4}{36}\right.\right.$$

ita bisa membuat sebuah fungsi dari hasil tersebut. Fungsi semacam itu sudah didefinisikan dalam file geometry.e milik Euler.

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

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

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

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

Hasilnya adalah rasional, yang tidak mudah untuk diperoleh jika kita menggunakan trigonometri klasikal.

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

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

Ini adalah sudut dalam bentuk derajat.

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

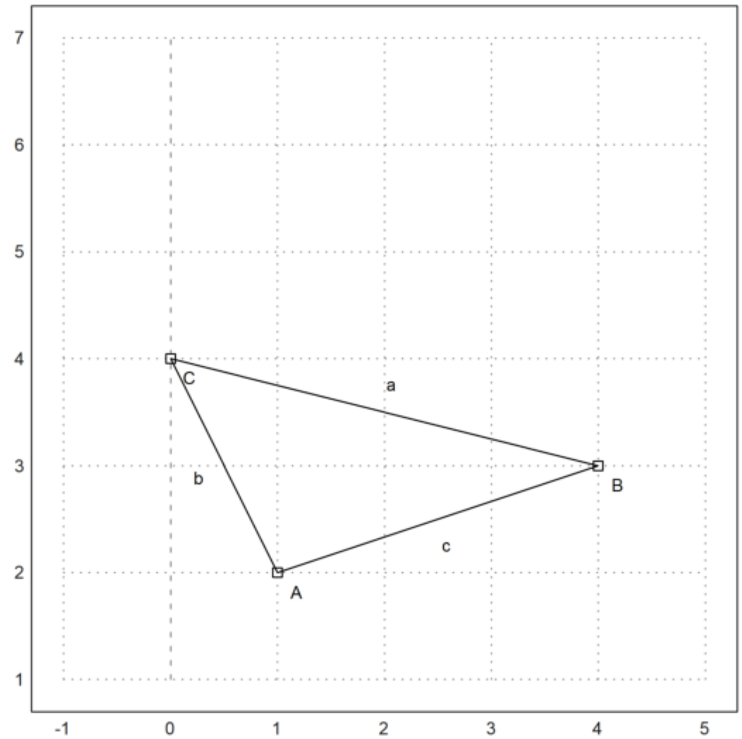
```
67°47'32.44''
```

Contoh Lain

Sekarang, mari kita coba sebuah contoh yang lebih rumit,

Kita mengatur tiga sudut segitiga seperti berikut.

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



Menggunakan Pythagoras, mudah untuk menghitung jarak antara dua titik. Pertama saya menggunakan fungsi distance dari file Euler untuk geometri. Fungsi distance ini menggunakan geometri klasik.

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

$$\sqrt{10}$$

Euler juga memiliki fungsi untuk menghitung quadrance antara dua titik.

Dalam contoh berikut, karena c+b tidak sama dengan a, egitiga ini bukan segitiga siku-siku.

```
>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 yang didasarkan pada hasil kali titik dua vektor. Hasilnya adalah beberapa pendekatan floating point.

$$A =< 1, 2 > \quad B =< 4, 3 >, \quad C =< 0, 4 >$$

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

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

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

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

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

```
32.4711922908
```

Dengan menggunakan kertas dan pensil, kita bisa melakukan hal yang sama dengan crosslaw. Kita masukkan kuadran a, b, dan c ke dalam crosslaw dan menyelesaikannya untuk x.

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

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

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

Artinya, itulah yang dilakukan oleh fungsi spread yang didefinisikan dalam ”geometry.e”.

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

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

Maxima mendapatkan hasil yang sama menggunakan trigonometri biasa, jika kita memaksanya. Ini memang menyelesaikan istilah sin(arccos(...)) tmenjadi hasil fraksional. Kebanyakan siswa mungkin tidak dapat melakukan ini.

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

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

Setelah kita mendapatkan spread di titik B, kita bisa menghitung tinggi ha pada sisi a. Ingat bahwa

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

secara definisi.

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

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

Gambar berikut telah dihasilkan dengan program geometri C.a.R., whyang dapat menggambar quadrance dan spread.

image: (20) Rational_Geometry_CaR.png

Secara definisi, panjang ha adalah akar kuadrat dari quadrancenya.

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

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

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

```
>$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);
```

Kita pertama-tama menghitung spread di titik B untuk segitiga dengan sisi a, b, dan c. Kemudian, kita menghitung luas kuadrat (apa itu "quadrea"?), memfaktorkannya dengan Maxima, dan kita mendapatkan rumus terkenal dari Heron.

Memang, ini sulit dilakukan dengan kertas dan pensil.

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

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

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

Aturan Triple Spread

Kekurangan dari spread adalah bahwa mereka tidak dapat dijumlahkan dengan cara yang sederhana seperti sudut.

Namun, ketiga spread dari sebuah segitiga memenuhi aturan "triple spread" berikut.

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

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

Aturan ini berlaku untuk sembarang tiga sudut yang jumlahnya 180°.

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

Karena spread dari

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

adalah sama, aturan triple spread juga berlaku, jika

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

Karena spread dari sudut negatif adalah sama, aturan triple spread juga berlaku, jika

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

Sebagai contoh, kita dapat menghitung spread dari sudut 60°. Nilainya adalah 3/4. ersamaan tersebut memiliki solusi kedua, di mana semua spread adalah 0.

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

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

Spread dari 90° jelas adalah 1. Jika dua sudut jumlahnya 90°, tspread mereka menyelesaikan persamaan triple spread dengan a,b,1. Dengan perhitungan berikut, kita mendapatkan a+b=1.

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

$$(y+x+1)^2=2\left(y^2+x^2+1\right)+4\,x\,y$$

$$[x=1-y]$$

Karena spread dari 180°-t sama dengan spread dari t, rumus triple spread juga berlaku jika salah satu sudut adalah jumlah atau selisih dari dua sudut lainnya.

Jadi, kita bisa mencari spread dari sudut yang digandakan. Perhatikan bahwa ada dua solusi lagi. Kita akan menjadikannya sebuah fungsi.

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

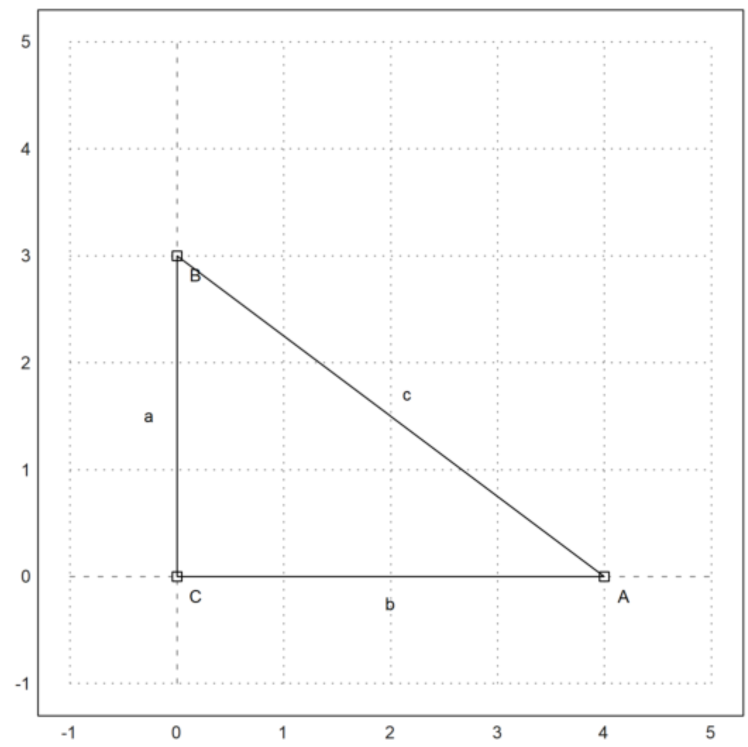
$$[x = 4\,a - 4\,a^2, x = 0]$$

```
- 4 (a - 1) a
```

Sudut Pembagi

Inilah situasinya, yang sudah kita ketahui.

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



Mari kita hitung panjang sudut pembagi di titik A. Namun, kita ingin menyelesaikannya untuk a,b, dan c secara umum.

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

Jadi, kita pertama-tama menghitung spread dari sudut yang dibagi di titik A, menggunakan rumus triple spread.

Masalah dengan rumus ini muncul lagi. Ada dua solusi. Kita harus memilih yang benar. Solusi lainnya merujuk pada sudut yang dibagi 180°-wa.

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

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

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

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

Mari kita periksa untuk the Egyptian rectangle.

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

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

Kita dapat mencetak sudut dalam Euler, setelah mengonversi spread ke radian.

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

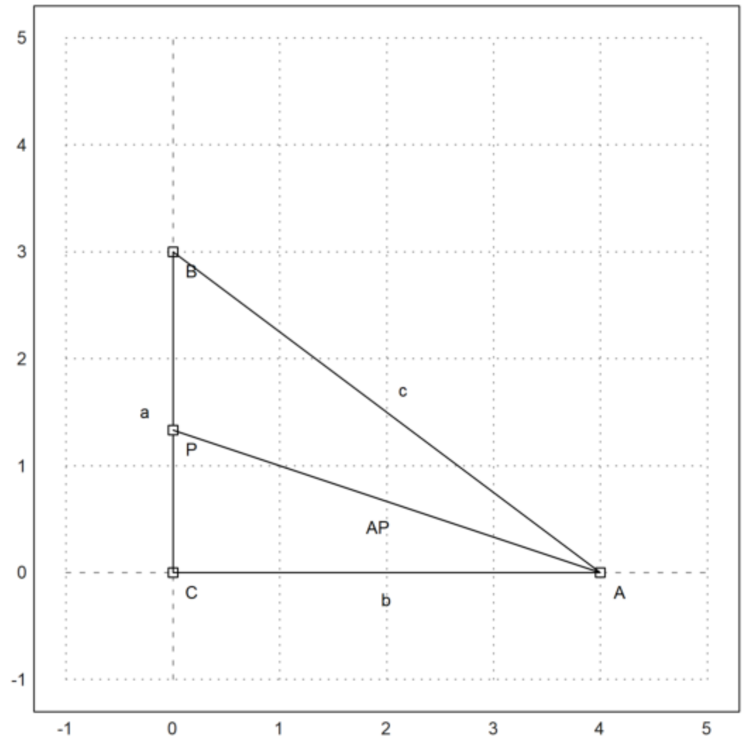
```
18°26'5.82''
```

Titik P adalah titik perpotongan antara sudut pembagi dengan sumbu y.

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

```
[0,  1.33333]
```

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



Mari kita periksa sudut-sudut dalam contoh spesifik kita.

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

```
0.321750554397
0.321750554397
```

Sekarang kita menghitung panjang bisektor AP.

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

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

berlaku di sembarang segitiga. Jika kita kuadratkan, ini akan diterjemahkan ke dalam apa yang disebut "hukum spread."

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

dimana a,b,c menunjukkan quadrance.
Karena spread CPA adalah 1-sa2, kita mendapatkan dari situ bisa/1=b/(1-sa2) dan dapat menghitung bisa (quadrance dari sudut bisektor).

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

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

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
```

We can also compute P using the spread formula.

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

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

Nilainya sama dengan yang kita peroleh menggunakan rumus trigonometri.

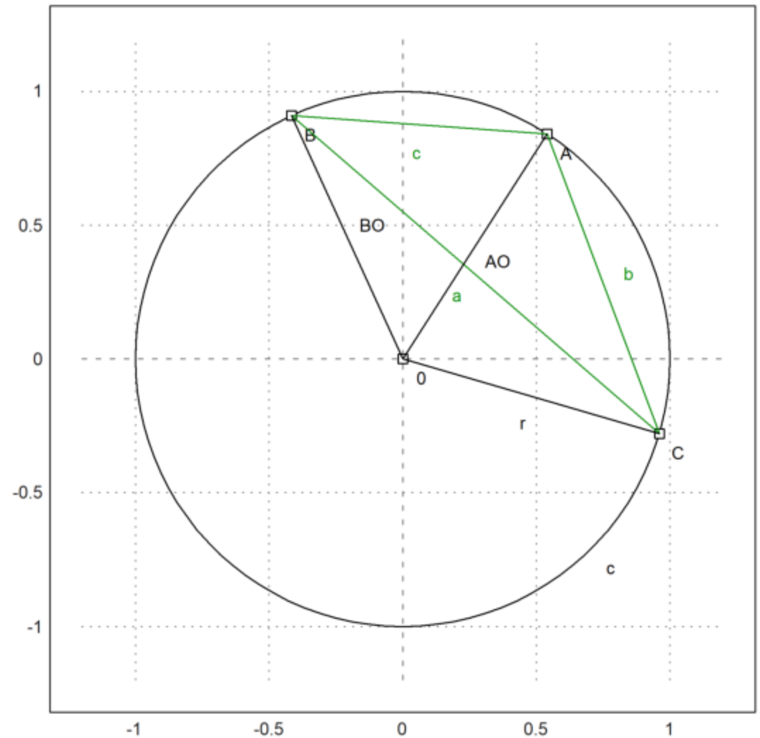
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

```
1.33333333333
```

Sudut Chord

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 triple spread untuk sudut di pusat O terhadap r. Dengan demikian, kita mendapatkan rumus untuk jari-jari kuadratik dari lingkaran luar dalam istilah quadrance dari sisi-sisinya.

Kali ini, Maxima menghasilkan beberapa nol kompleks, yang akan 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
```

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

Kita bisa menjadikannya sebuah fungsi Euler.

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

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

```
>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, spread tersebut 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
```

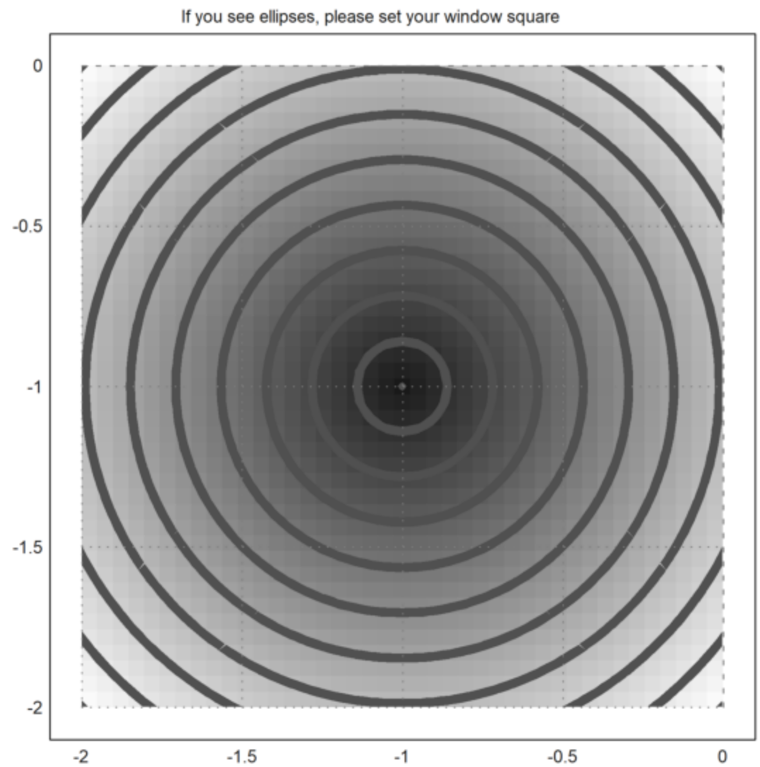
$$0$$

Contoh 6: Jarak Minimal pada Bidang

Keterangan Awal

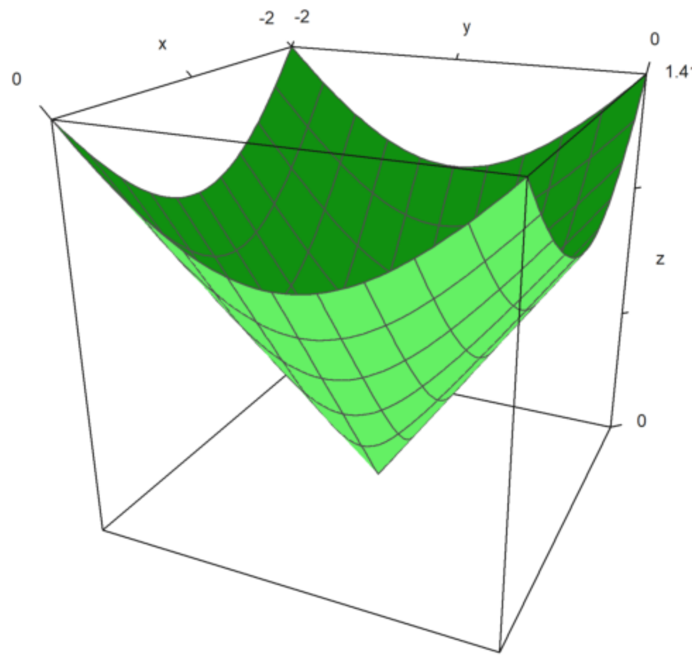
Fungsi yang mengaitkan sebuah titik M di bidang dengan jarak AM antara titik tetap A dan M, memiliki garis level yang cukup sederhana: lingkaran yang berpusat di A.

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



Dan grafiknya juga cukup sederhana: bagian atas sebuah kerucut.

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

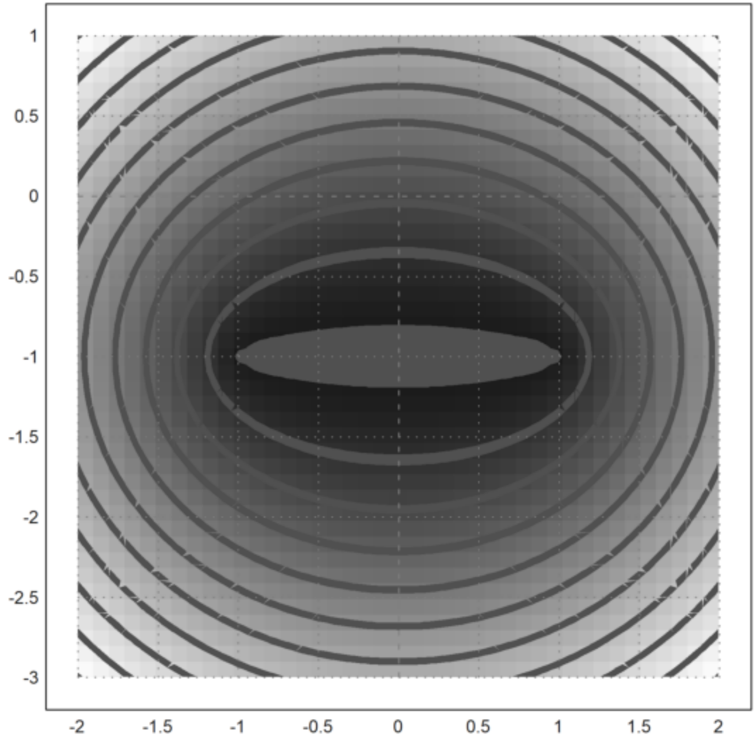


Tentu saja, nilai minimum 0 dicapai di titik A.

Dua Titik

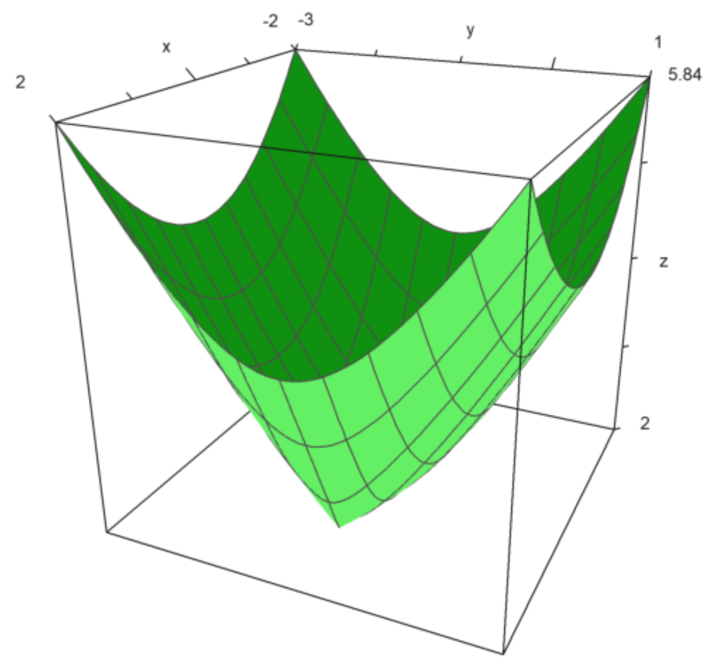
Sekarang kita melihat fungsi MA+MB dimana A dan B adalah dua titik (tetap). Ini adalah "fakta yang sudah dikenal" bahwa kurva levelnya adalah elips, dengan titik fokus di A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

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



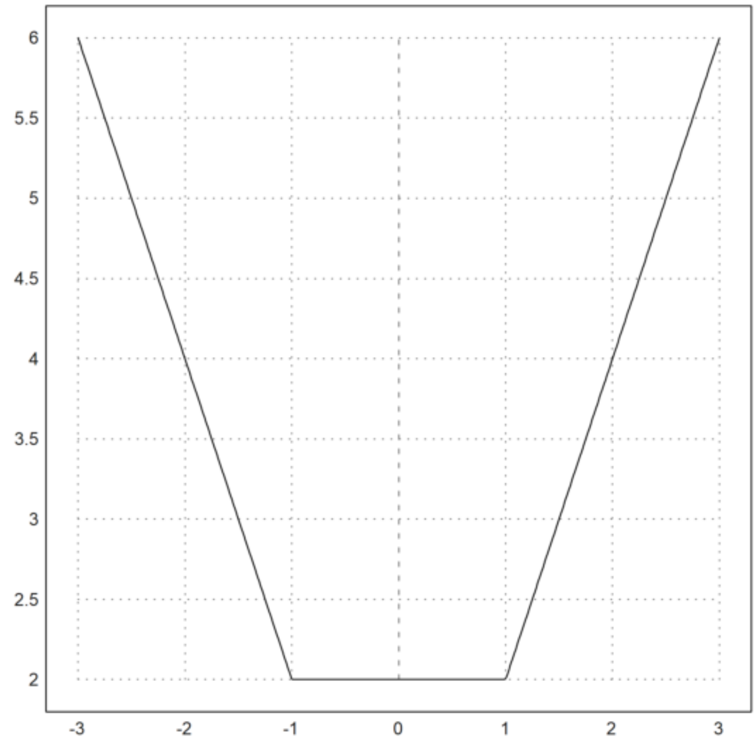
Grafiknya lebih menarik:

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



Pembatasan pada garis (AB) lebih terkenal:

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

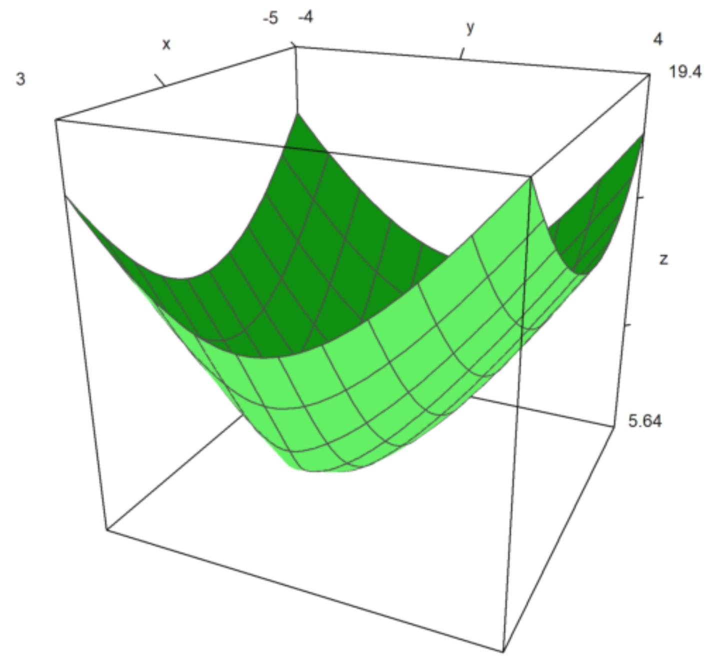


Sekarang hal-hal menjadi kurang sederhana: Sedikit kurang dikenal bahwa MA+MB+MC mencapai minimum di satu titik di bidang, tetapi untuk menentukannya tidaklah sederhana:

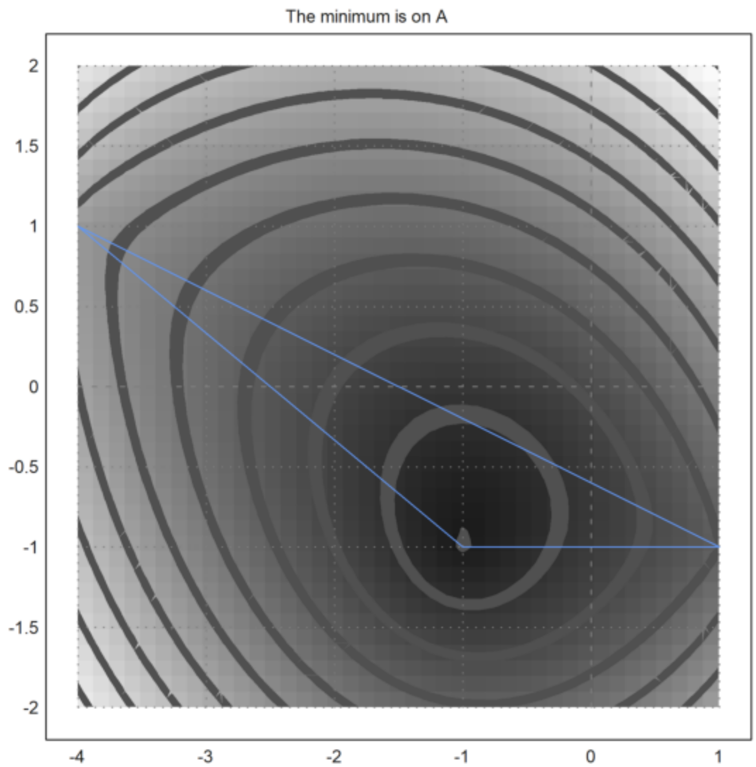
1) Jika salah satu sudut segitiga ABC lebih dari 120° (katakanlah di A),maka minimum dicapai di titik itu sendiri (katakanlah AB+AC).

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;
```

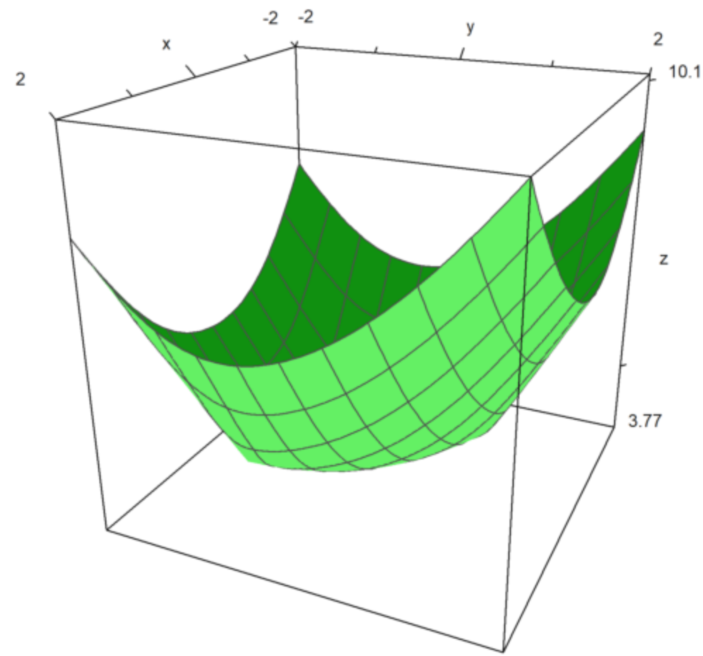



```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

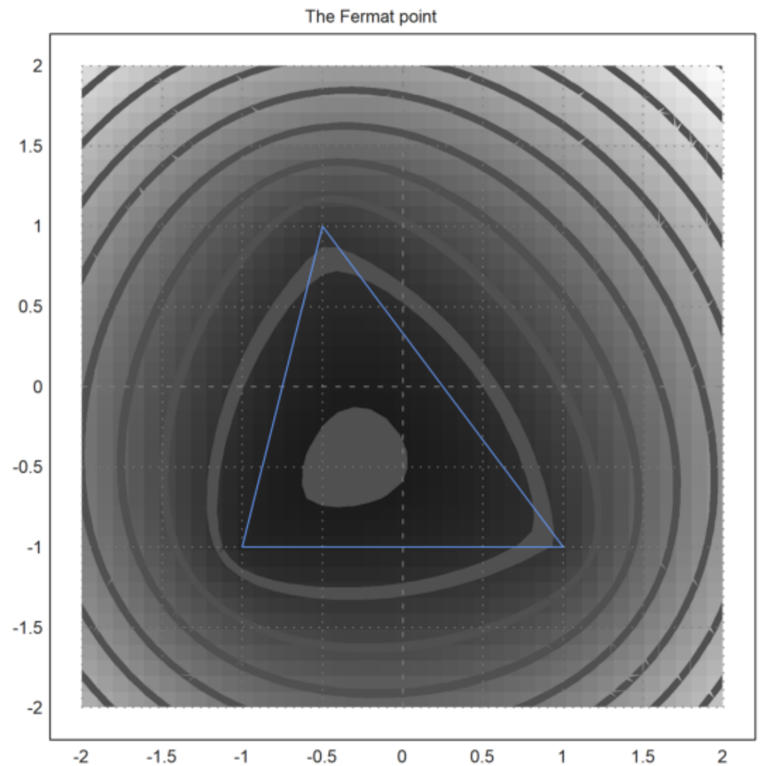


2) Namun, jika semua sudut segitiga ABC kurang dari 120°, minimum berada di titik F di dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (kemudian 120° masing-masing):

```
>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;
```

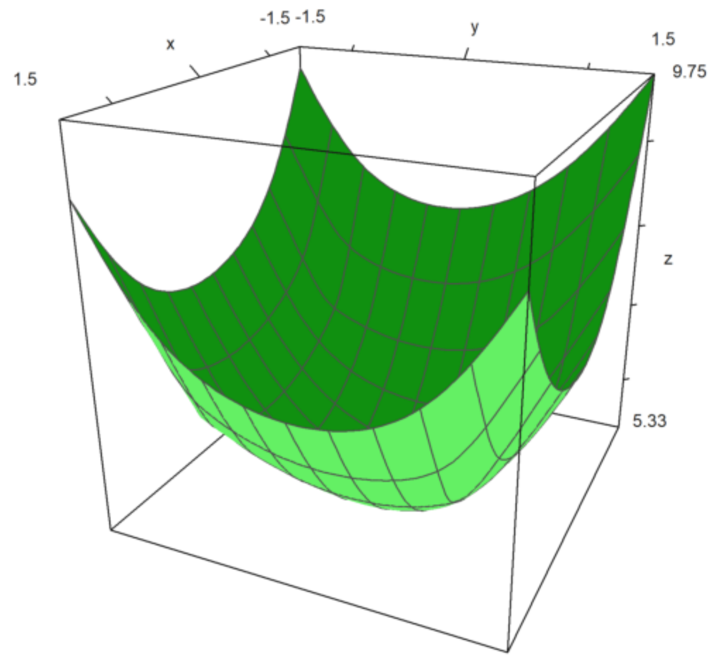


Merupakan aktivitas yang menarik untuk menggambar gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu sebuah perangkat lunak yang ditulis dalam Java yang memiliki instruksi ”contour lines” ...

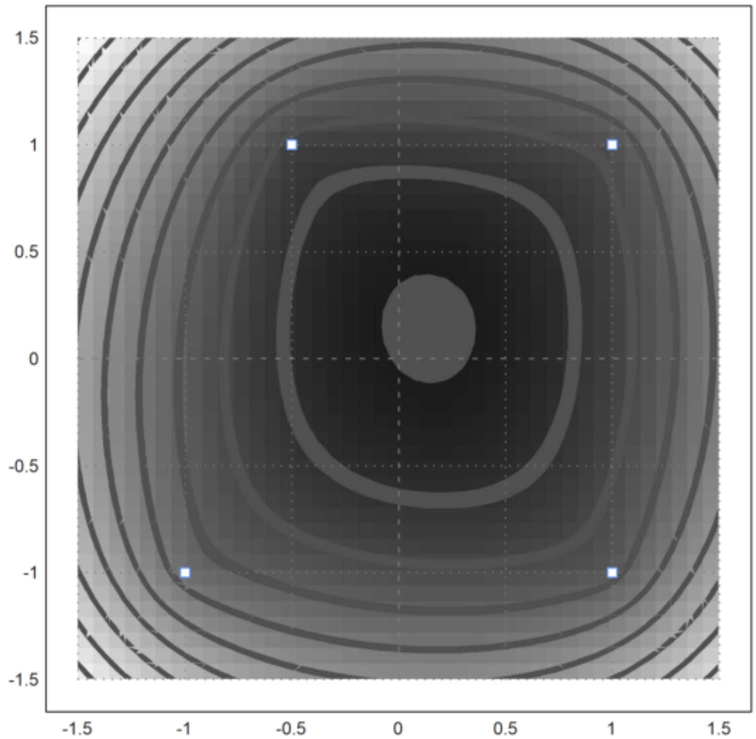
Semua ini ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; dia menulis surat kepada dilettantes lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di pajak penghasilan. Jadi, titik unik F sedemikian sehingga FA+FB+FC minimal, disebut titik Fermat dari segitiga. Namun, tampaknya beberapa tahun sebelumnya, orang Italia Torricelli telah menemukan titik ini sebelum Fermat! Bagaimanapun, tradisinya adalah menandai titik ini dengan F...

Langkah selanjutnya adalah menambahkan titik ke-4 D adan mencoba meminimalkan MA+MB+MC+MD; misalnya, Anda adalah operator TV kabel dan ingin mengetahui di mana Anda harus menempatkan antena Anda agar 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 itu tidak dicapai di salah satu dari titik sudut A, B, C atau D:

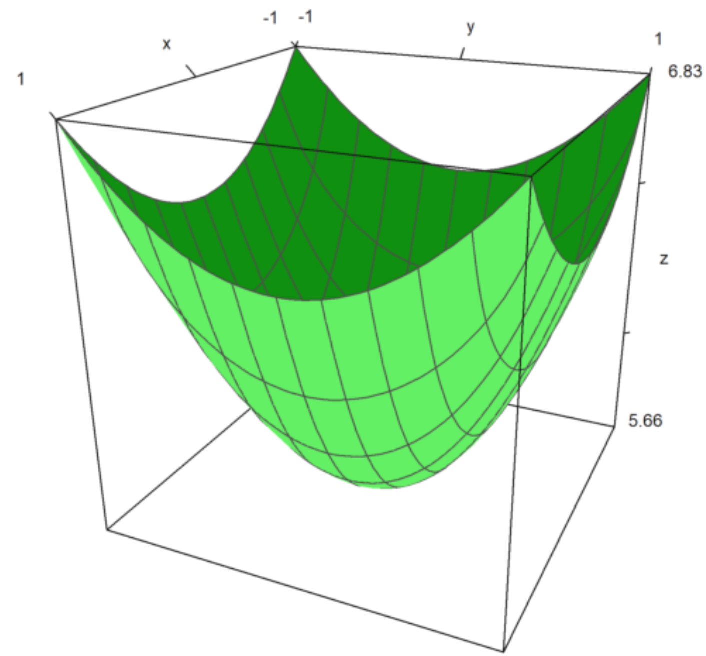
```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```

```
[0.142858,  0.142857]
```

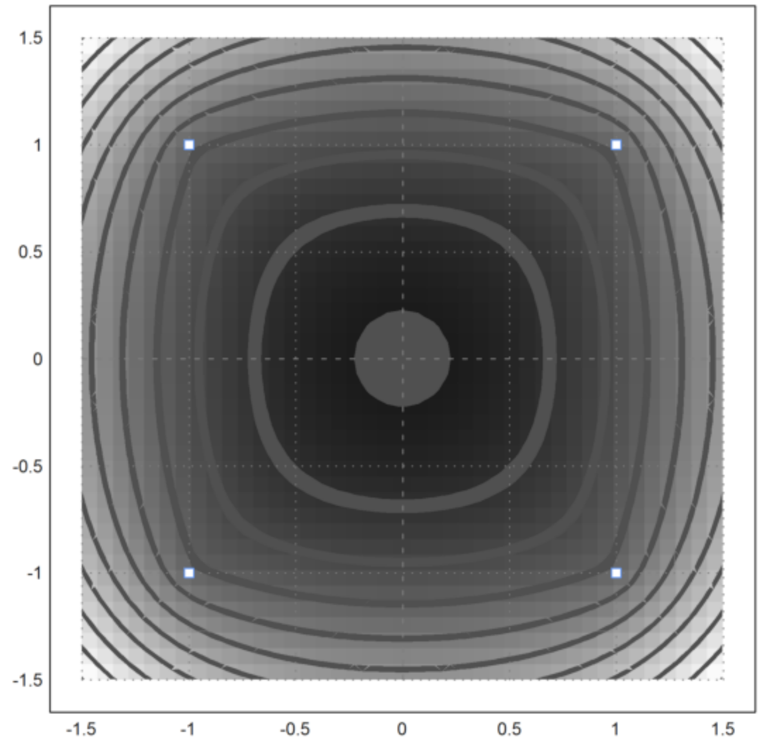
Sepertinya dalam kasus ini, koordinat dari titik optimal adalah rasional atau mendekati rasional...

Sekarang, jika ABCD adalah sebuah persegi, kita berharap bahwa titik optimal akan berada di 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;
```


1.5
1
0.5
0
-0.5
-1
-1.5
-1.5
-1
-0.5
0
0.5
1
1.5

Contoh 7: Bola Dandelin dengan Povray

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

Pertama, kita menghitung jari-jari bola.

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

Kita menggunakan file geometry.e dari Euler untuk ini.

```
>load geometry;
```

Pertama, kita perlu dua garis yang membentuk kerucut.

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

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

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

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

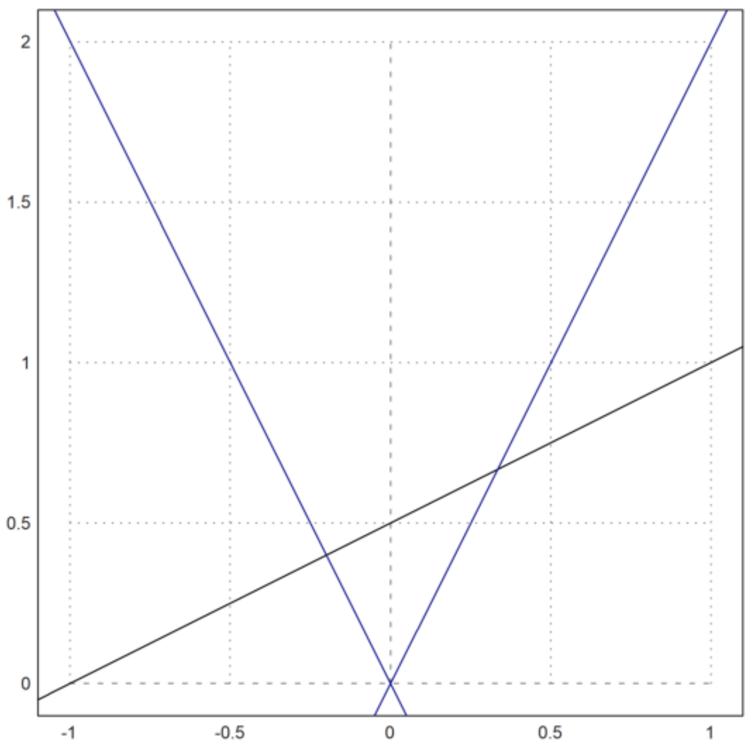
Kemudian, kita tambahkan garis ketiga.

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

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

Kita gambarkan semua yang telah ada sejauh ini.

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



Sekarang kita ambil titik umum di sumbu-y.

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

```
[0, u]
```

Hitung jarak ke g1.

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

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

Hitung jarak ke g.

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

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

Dan temukan pusat dari dua lingkaran, di mana jaraknya sama.

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

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

Ada dua solusi.

Kita evaluasi solusi simbolik, dan menemukan kedua pusat serta kedua jaraknya.

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

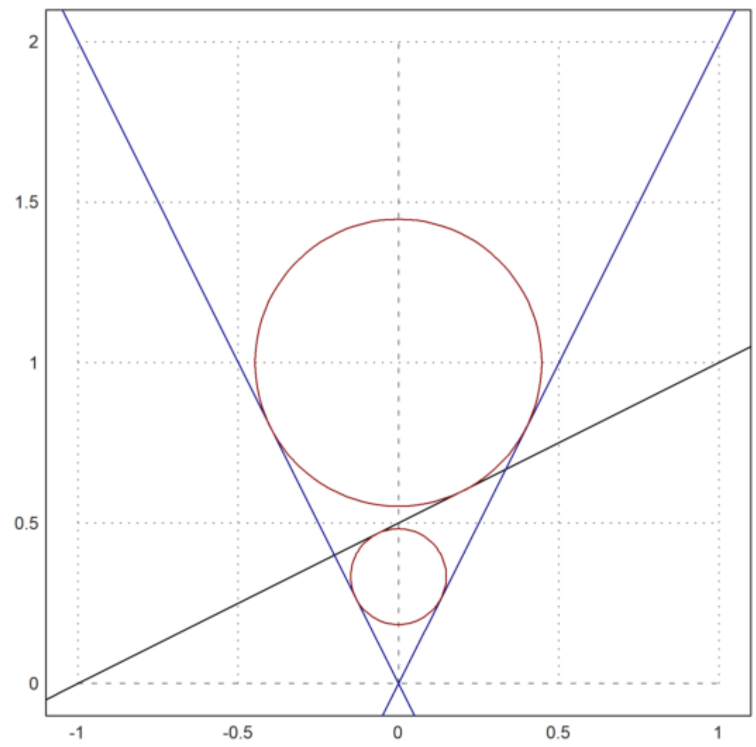
```
[0.333333,  1]
```

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

```
[0.149071,  0.447214]
```

Gambarkan lingkaran-lingkaran ke dalam gambar.

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



2
1.5
1
0.5
0
-1
-0.5
0
0.5
1

Selanjutnya, kita gambarkan semuanya dengan Povray. Perhatikan bahwa Anda dapat mengubah perintah apa pun dalam urutan perintah Povray berikut, dan menjalankan kembali semua perintah dengan Shift-Return.

Pertama, kita muat fungsi-fungsi Povray.

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

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

Kita atur adegan dengan tepat.

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

Selanjutnya, kita tulis dua bola ke dalam file Povray.

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

Dan kerucut, dalam keadaan transparan.

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

Kita menghasilkan sebuah bidang yang dibatasi oleh 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, di mana bola-bola menyentuh kerucut.

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

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

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

Selanjutnya, kita hitung interseksi antara 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)));
```

Kita menghubungkan titik-titik tersebut dengan segmen garis.

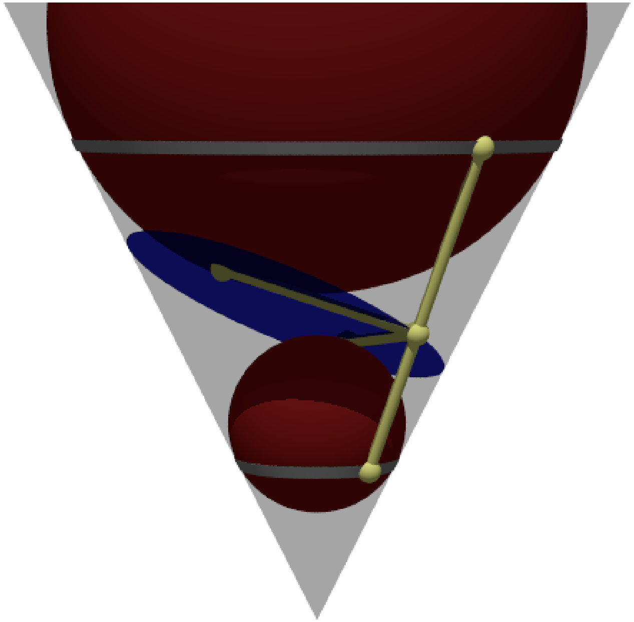
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan pita abu-abu, di mana bola-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)));
```

Jalankan program Povray.

```
>povend();
```



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

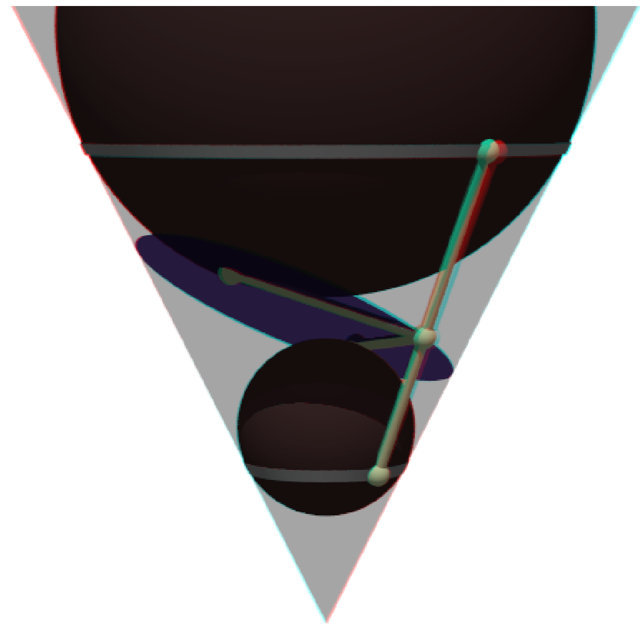
```
>function scene () ...
```

```
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g();
```

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

Anda memerlukan kacamata merah/cyan untuk menghargai efek berikut.

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



Contoh 8: Geometri Bumi

Dalam notebook ini, kita 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, kita menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, dengan nilai negatif untuk selatan dan barat). Berikut adalah koordinat untuk Kampus FMIPA UNY.

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

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

Anda bisa mencetak posisi ini menggunakan fungsi 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 kota ke kota lainnya di atas bola ideal. Vektor ini adalah [heading,distance] dalam radian. Untuk menghitung jarak di permukaan bumi, kita kalikan dengan jari-jari bumi pada 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 pendekatan yang baik. Rutinitas berikut menggunakan pendekatan yang bahkan lebih baik. Pada jarak yang begitu pendek, hasilnya hampir sama.

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

```
88.0114026318km
```

Hasil di atas merupakan perkiraan jarak FMIPA-Semarang

Ada fungsi untuk menghitung heading, yang memperhitungkan bentuk elips bumi. Sekali lagi, kita mencetak hasilnya dengan cara yang lebih canggih.

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

```
     65.34°
```

Sudut segitiga melebihi 180° di atas permukaan bola.

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

```
 180°0'10.77''
```

ni 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"
```

```
 2116.02948749 km^2
```

Hasil di atas adalah perkiraan luas segitiga FMIPA-Solo-Semarang

Ada fungsi untuk ini, yang menggunakan 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 mengandung heading dan jarak, keduanya dalam radian. Untuk menambahkan vektor ke posisi, kita menggunakan "svector".
ntuk menambahkan vektor ke posisi, kita menggunakan "saddvector".

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

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

Fungsi-fungsi ini mengasumsikan sebuah bola ideal. Hal yang sama berlaku pada 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, yaitu 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 429.66km. Kita mendapatkan pendekatan yang baik.

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

```
431.565659488 km
```

Di atas merupakan perkiraan jarak Tugu Jogja - Monas Jakarta

Heading-nya sama dengan yang dihitung di Google Earth.

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

```
294°17’2.85’’
```

Namun, kita tidak lagi mendapatkan posisi target yang tepat jika kita menambahkan heading dan jarak ke posisi asal. Hal ini terjadi karena kita tidak menghitung fungsi invers dengan tepat, melainkan mengambil pendekatan jari-jari bumi sepanjang jalur.

```
>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 saja, kita tidak dapat berlayar dengan heading yang sama dari satu tujuan ke tujuan lain jika kita ingin mengambil jalur terpendek. Bayangkan, Anda terbang ke arah NE mulai dari titik mana pun di bumi. Maka Anda akan berputar ke arah kutub utara. Lingkaran besar tidak mengikuti heading yang konstan!

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

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

Sekarang kita menambahkan 10 kali satu persepuluh dari jarak, menggunakan heading ke Monas yang kita peroleh di Tugu.

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

Hasilnya jauh dari yang diharapkan.

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

```
 S 6°11.250' E 106°48.372'
      1.529km
```

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

```
>P1=[30°,10°]; P2=[30°,50°];
```

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

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

```
79.69°
```

Namun, jika kita mengikuti bacaan kompas ini, kita akan berputar ke arah kutub utara! Jadi, kita harus menyesuaikan heading kita sepanjang perjalanan. Untuk keperluan kasar, kita 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
```

Kita mendapatkan pendekatan yang baik jika kita menyesuaikan heading kita 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 deret posisi GPS sepanjang lingkaran besar menuju Monas dengan fungsi navigate.

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

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

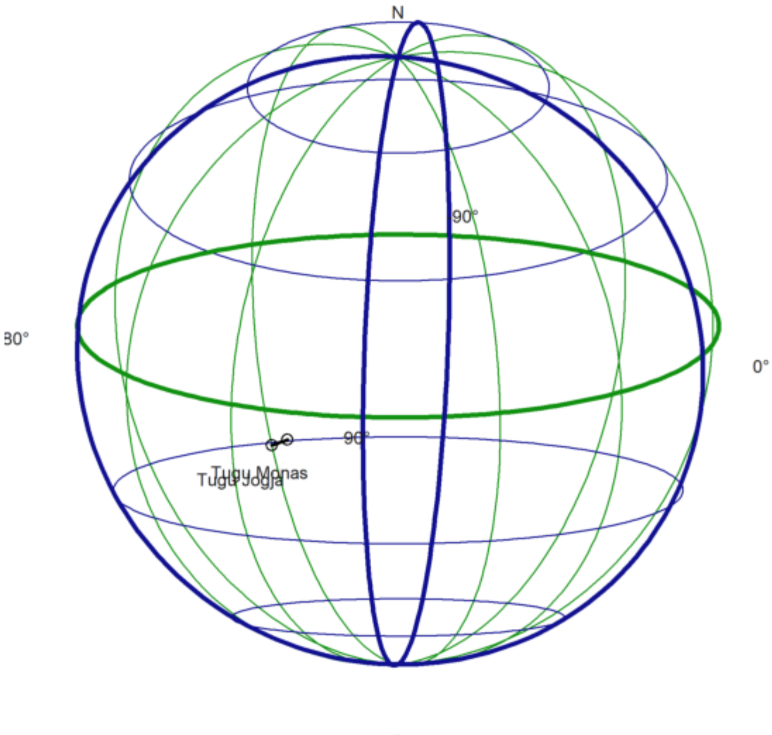
Kita menulis sebuah fungsi yang menggambarkan bumi, kedua posisi, dan posisi di antara keduanya.

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

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

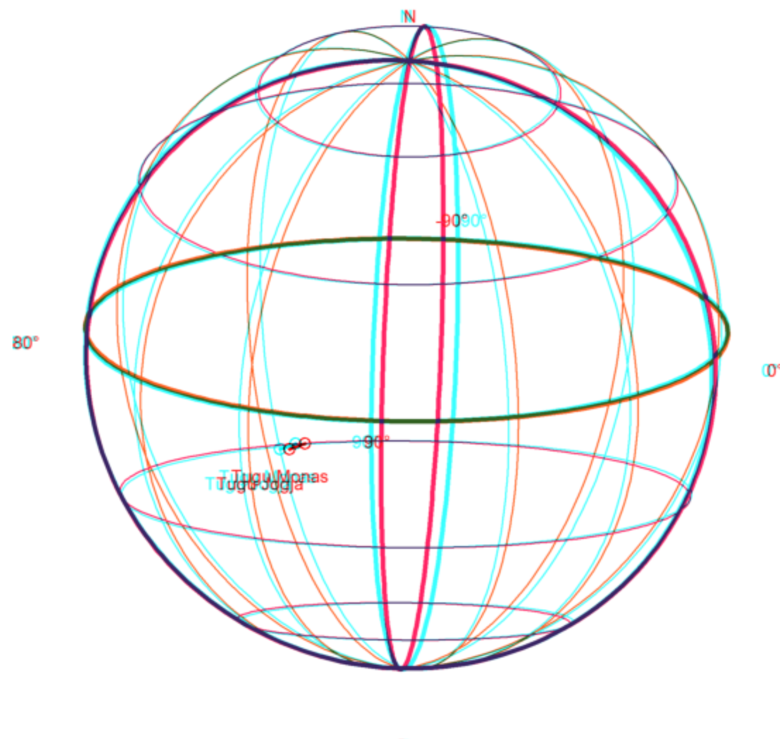
Sekarang gambarkan semuanya.

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



Or use plot3d to get an anaglyph view of it. This looks really great with red/cyan glasses.

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


N
-90°
80°
0°
90°
Tugu Monas
Tugu Jogja

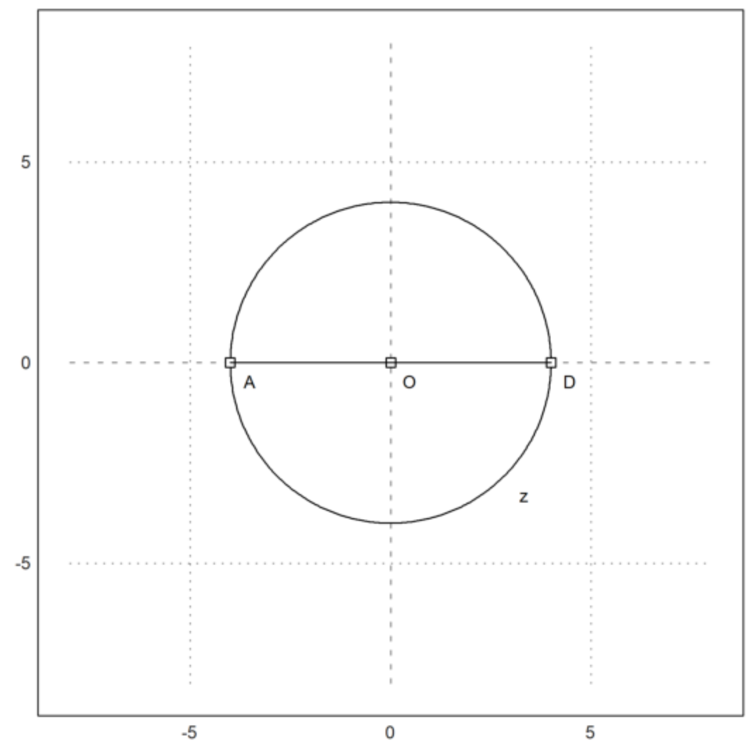
1. Gambarlah segi-n beraturan jika diketahui titik pusat O, n, dan jarak titik pusat ke titik-titik sudut segi-n tersebut (jari-jari lingkaran luar segi-n), r.

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.

JAWAB :
Saya akan mencoba membuat segi-6 beraturan dengan jari-jari 4
- membuat lingkaran dengan pusat O(0,0) dan jari jari 4

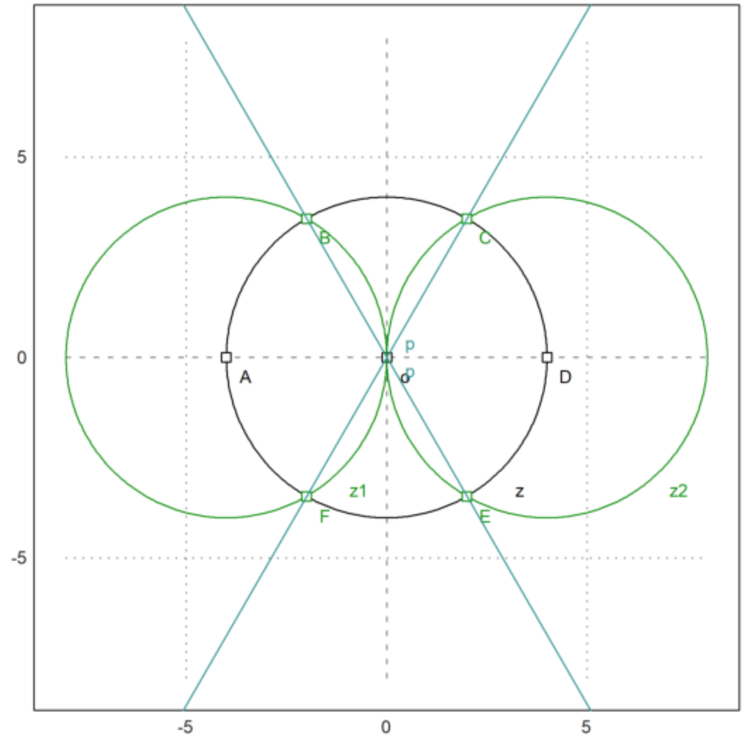
```
>o &:= [0,0]; z=circleWithCenter(o,4);
>color(1); setPlotRange(8); plotPoint(O); plotCircle(z);
>A=[-4,0]; plotPoint(A,"A");
>D=[4,0]; plotPoint(D,"D");
>plotSegment(A,D,""):
```



- membuat 2 lingkaran di sisi kanan dan kiri yang memalui O dan berjari jari 4 juga

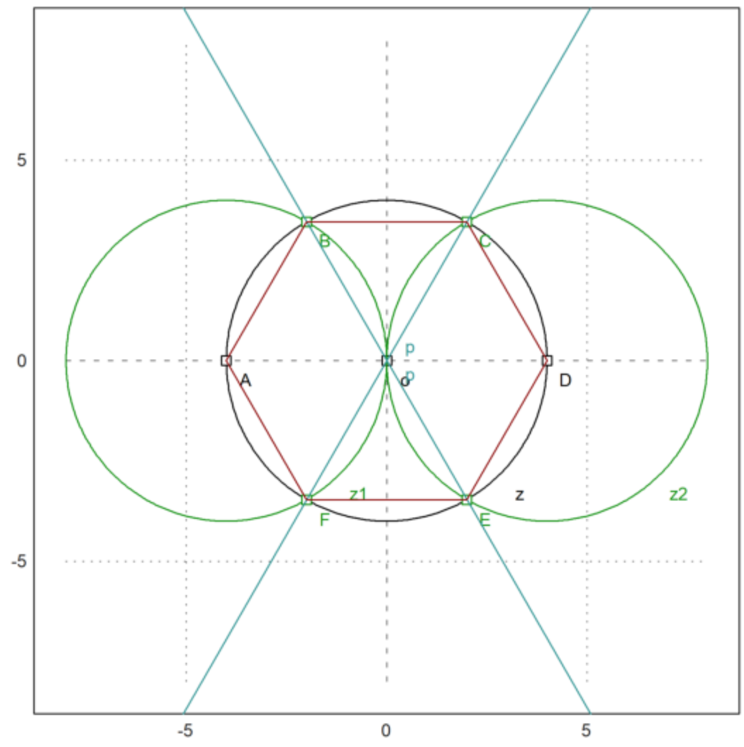
```
>z1=circleWithCenter(A,distance(A,o));
>z2=circleWithCenter(D,distance(D,o));
>B=circleCircleIntersections(z1,z);
>C=circleCircleIntersections(z,z2);
```

```
>E=circleCircleIntersections(z2,z);
>F=circleCircleIntersections(z,z1);
>r=lineThrough(B,h); s=lineThrough(g,i);
>setPlotRange(8); plotPoint(o); plotCircle(z); plotPoint(A,"A"); plotPoint(D,"D"); plotSegment(D,D,"
>color(3); plotCircle(z1); plotCircle(z2); plotPoint(B); plotPoint(C); plotPoint(E); plotPoint(F);
>color(5); plotLine(r); plotLine(s):
```



- Berikut ini gambar segi-6 beraturan ABCDEF (berwarna merah) yang dihasilkan dari proses di atas

```
>color(2); plotSegment(A,B,""); plotSegment(A,F,""); plotSegment(B,C,""); ...
> plotSegment(C,D,""); plotSegment(D,E,""); plotSegment(E,F,""):
```

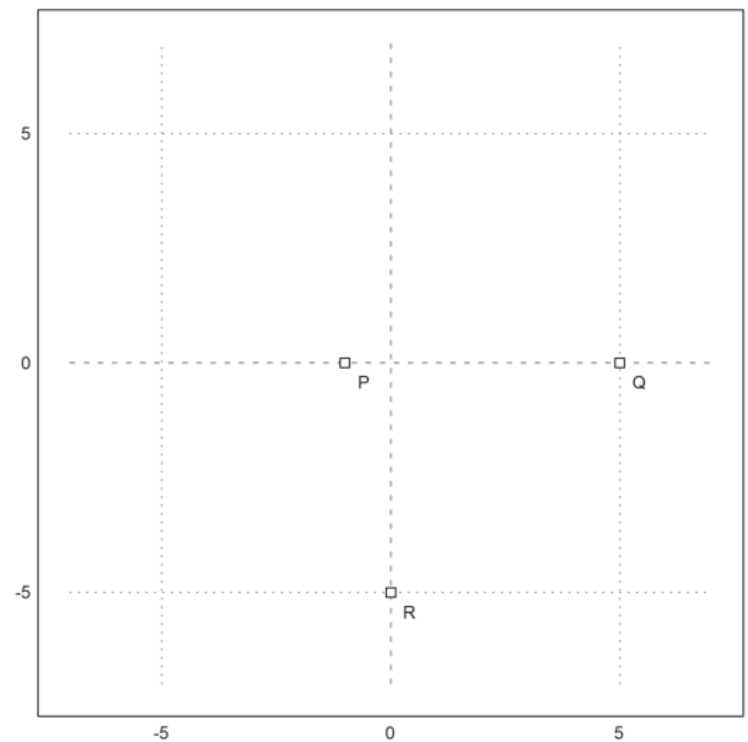


2. Gambarlah suatu parabola yang melalui 3 titik yang diketahui.

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.

JAWAB :
- menentukan 3 titik yang dilewati paraabola

```
>setPlotRange(7); P=[-1,0]; Q=[5,0]; R=[0,-5];
>plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R"):
```



- substitusi titik P, Q, R ke persamaan y=ax^2+bx+c, diperoleh 3 persamaan:

$$0 = 25a + 5b + c$$

$$0 = a - b + c$$

$$-5 = c$$

solusinya adalah :

```
>sol &= solve([25*a+5*b=-c,a-b=-c,c=-5],[a,b,c])
```

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

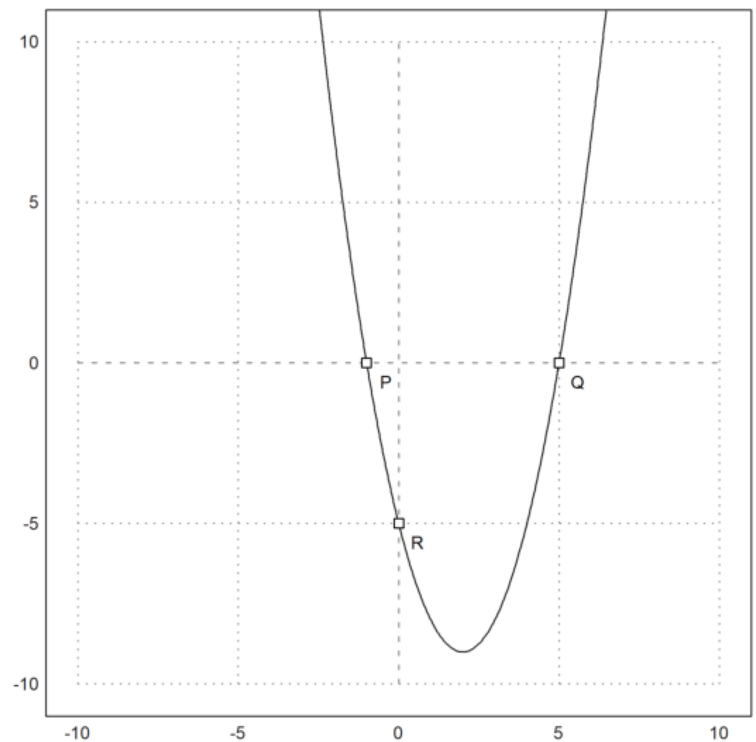
sehingga diperoleh persamaan kuadrat berikut :

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

```
 2
x  - 4 x - 5
```

Berikut ini plot kurva yang melalui titik P, Q, R :

```
>plot2d("x^2-4*x-5",-10,10,-10,10); plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R"):
```



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-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.

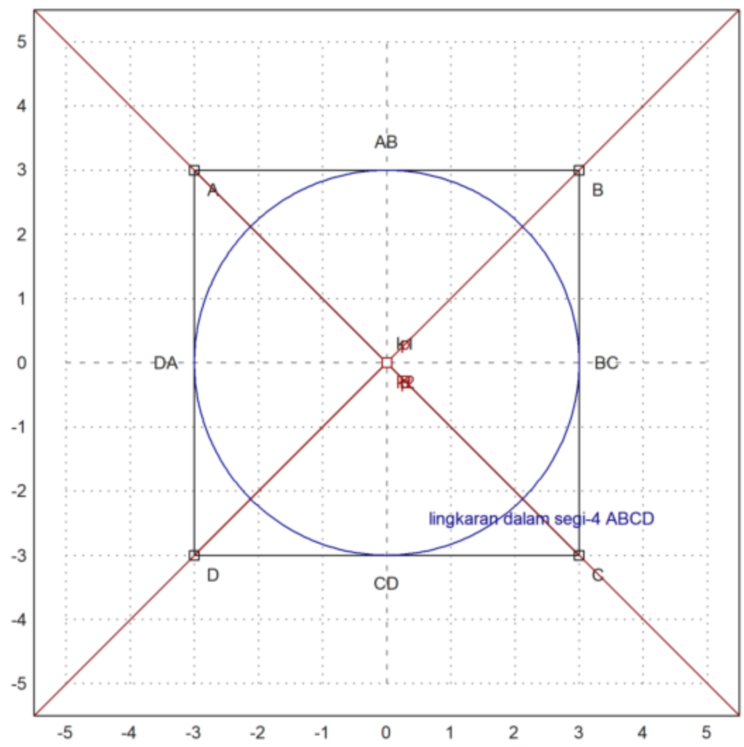
JAWAB :

```
>setPlotRange(-5,5,-5,5);
>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(D,A);
>plotSegment(A,C,"k1"); color(2);
>plotSegment(B,D,"k2"); color(2);
>k1=lineThrough(A,C);
>k2=lineThrough(B,D);
```

```
>p=lineIntersection(k1,k2);
>plotLine(k1); plotLine(k2);
>plotPoint(p, "P");
>r=norm(p-projectToLine(p,lineThrough(A,B)))
```

3

```
>plotCircle(circleWithCenter(p,r),"lingkaran dalam segi-4 ABCD"); color(5):
```



Dari gambar di atas dapat disimpulkan bahwa segiempat ABCD merupakan segiempat garis singgung lingkaran dalam segiempat tersebut, serta garis bagi keempat sudutnya bertemu di titik P

```
>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

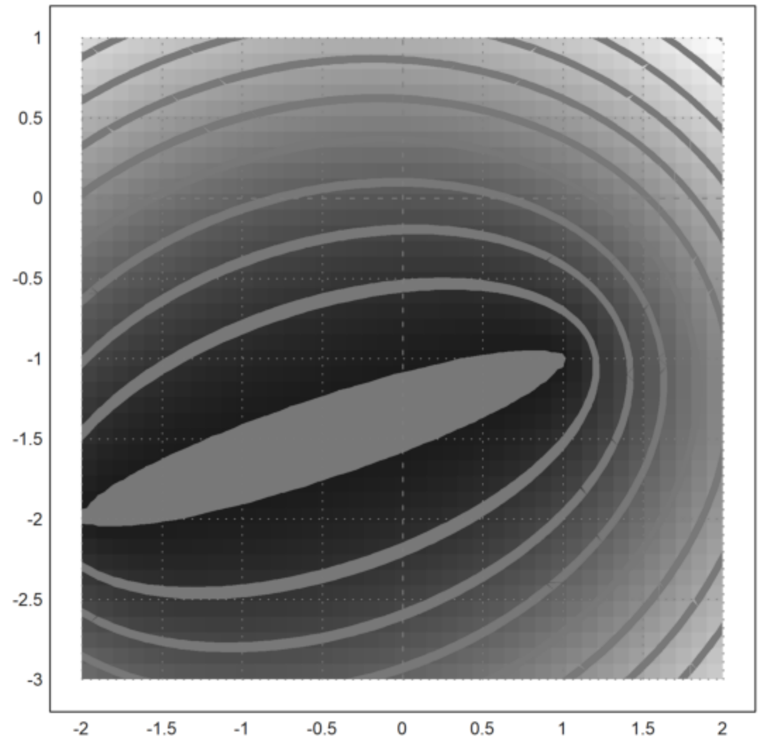
```
>AD.BC
```

36

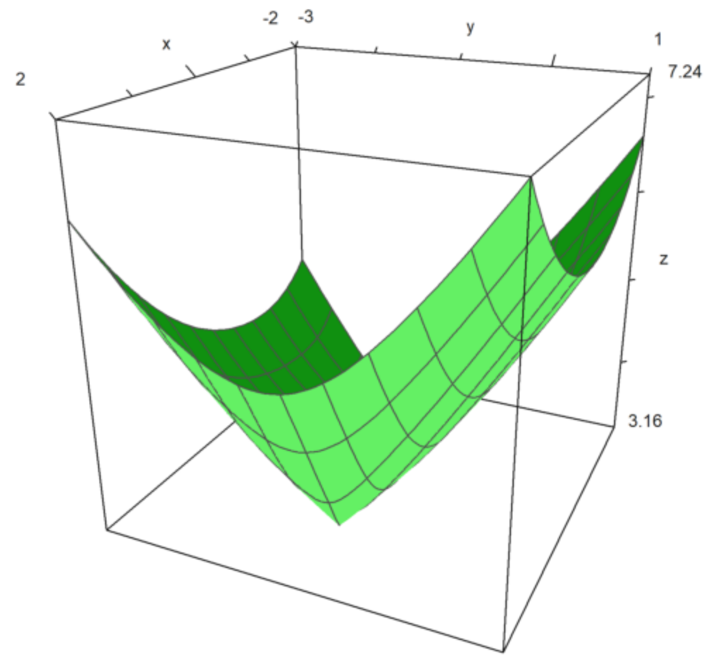
Terbukti bahwa hasil kali panjang sisi yang berhadapan (AB-CD dan AD-BC) sama yaitu 36

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

```
>P=[-2,-2]; 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):
```

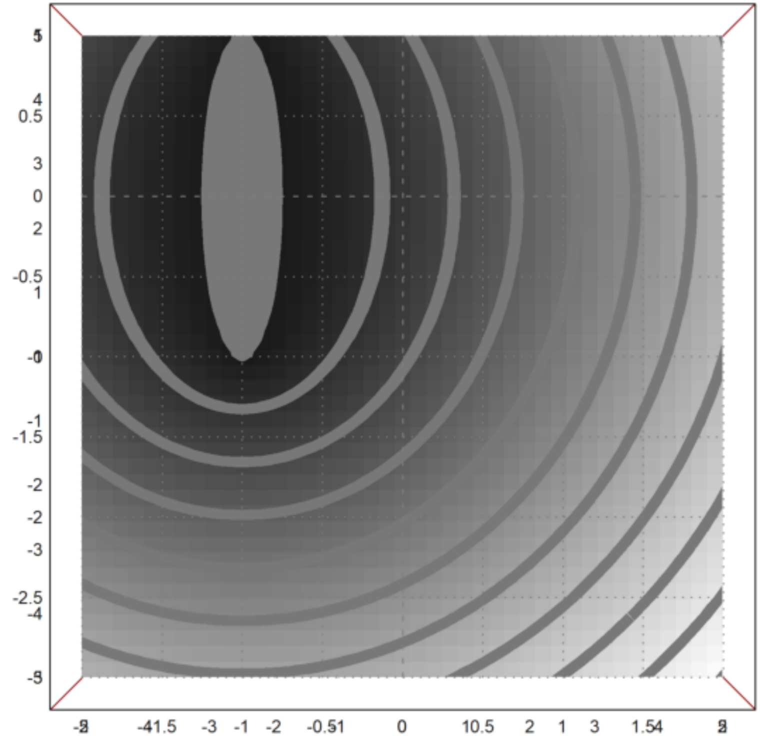



```
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```

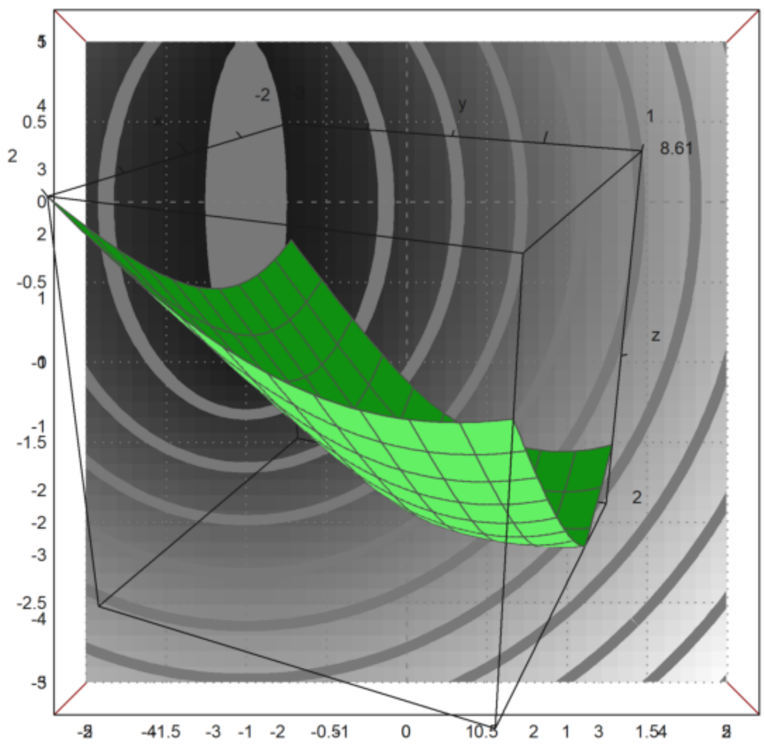


5. Gambarlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

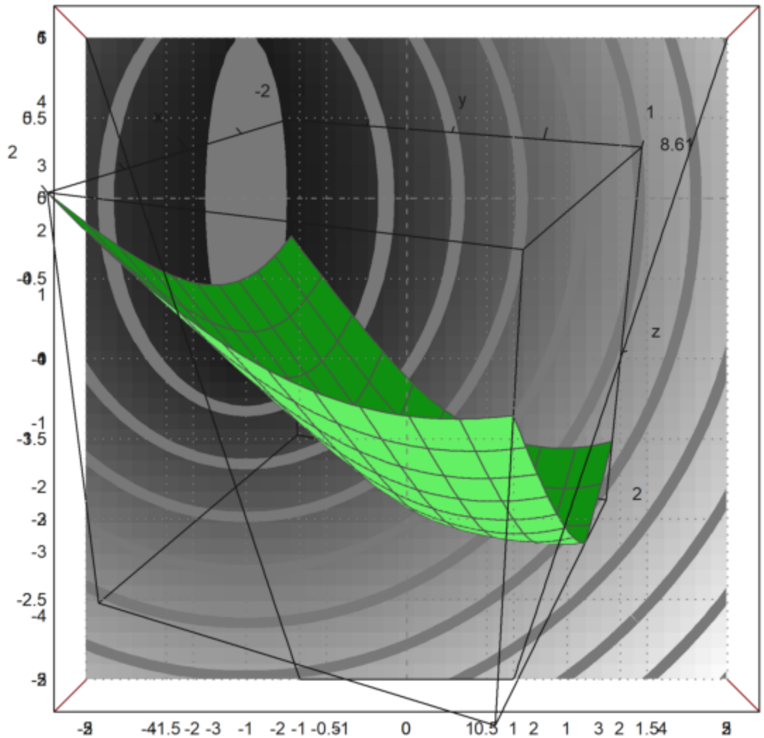
```
>P=[-1,-1]; Q=[1,-1];
>function d1(x,y):=sqrt((x-p[1])^2+(y-p[2])^2)
>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)+sqrt((x+Q[1])^2+(y+Q[2])^2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



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




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


y
z