

TI_EMT_Geometri_Prima

Rosalina_24030130035

Nama: Prima Rosalina
NIM: 24030130035
Kelas: Pendidikan Matematika C 24

Visualisasi dan Perhitungan Geometri dengan EMT

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

```
>load geometry
```

Numerical and symbolic geometry.

Fungsi-fungsi Geometri

Fungsi-fungsi untuk Menggambar Objek Geometri:

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

koordinat

```
setPlotRange(r): pusat bidang koordinat (0,0) dan batas-batas sumbu-x dan y adalah -r sd r  
plotPoint (P, "P"): menggambar titik P dan diberi label "P"  
plotSegment (A,B, "AB", d): menggambar ruas garis AB, diberi label "AB" sejauh d  
plotLine (g, "g", d): menggambar garis g diberi label "g" sejauh d  
plotCircle (c,"c",v,d): Menggambar lingkaran c dan diberi label "c"  
plotLabel (label, P, V, d): menuliskan label pada posisi P
```

Fungsi-fungsi Geometri Analitik (numerik maupun simbolik):

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

$ax+by=c$.

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

Fungsi-fungsi Khusus Untuk Geometri Simbolik:

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

y dengan titik A pada

sisi positif (kanan/atas) garis

quad(A,B): kuadrat jarak AB

spread(a,b,c): Spread segitiga dengan panjang sisi-sisi a,b,c, yakni $\sin(\alpha)^2$ dengan α 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 2ϕ , 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 plotkan mereka.

```
>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 mencakup 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]
```

Hitung garis tegak lurus melalui A pada BC.

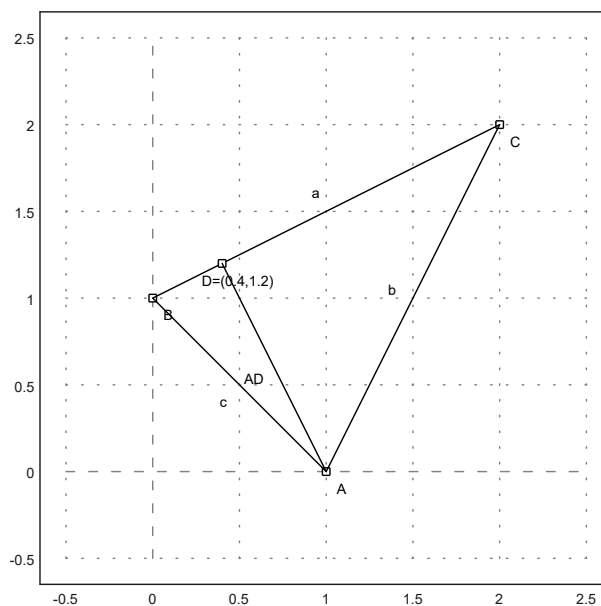
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan titik potongnya dengan BC.

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

Plotkan 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)
```

```
1.5
```

Bandingkan dengan rumus determinan.

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

```
1.5
```

Cara lain menghitung luas segitiga ABC:

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

1.5

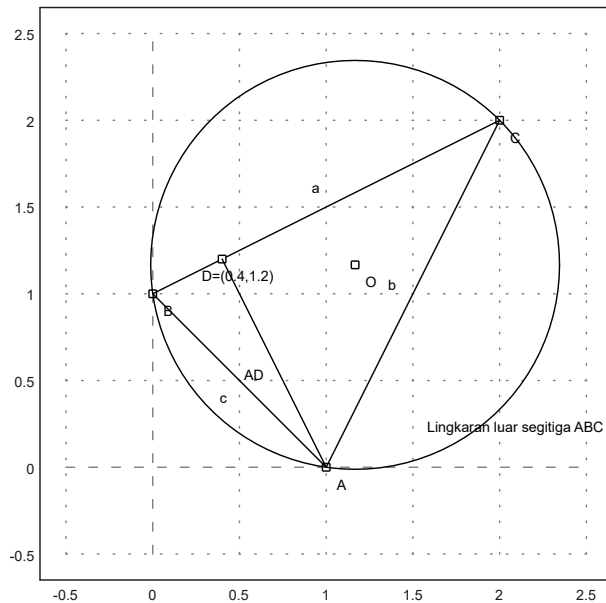
Sudut di C.

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

36°52'11.63''

Sekarang lingkaran keliling segitiga itu.

```
>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
```

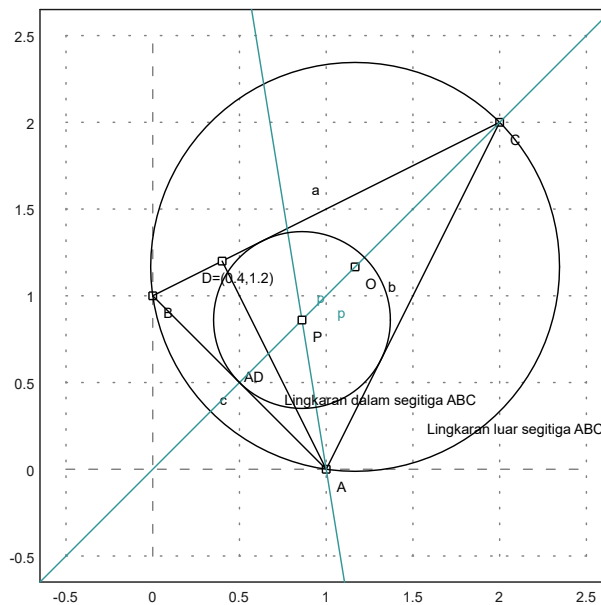
```
[0.86038, 0.86038]
```

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.
2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?
3. Hitung luas segitiga tersebut.
4. Tunjukkan bahwa garis bagi sudut yang ke tiga juga melalui titik pusat lingkaran dalam.
5. Gambar jari-jari lingkaran dalam.
6. Hitung luas lingkaran luar dan luas lingkaran dalam segitiga ABC. Adakah hubungan antara luas kedua lingkaran tersebut dengan luas segitiga ABC?

```
>load geometry
```

Numerical and symbolic geometry.

```
>setPlotRange(-2,6,-2,6); // mendefinisikan bidang koordinat baru
>A=[2,0]; plotPoint(A,"A"); // definisi dan gambar tiga titik
>B=[-1,2]; plotPoint(B,"B");
>C=[4,5]; plotPoint(C,"C");
>plotSegment(A,B,"c"); // c=AB
>plotSegment(B,C,"a"); // a=BC
>plotSegment(A,C,"b"); // b=AC
```

A = (2,0)
B = (-1,2)
C = (4,5)

$$AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

$$AB = \sqrt{(-1 - 2)^2 + (2 - 0)^2} = \sqrt{13}$$

$$BC = \sqrt{(4 - (-1))^2 + (5 - 2)^2} = \sqrt{34}$$

$$CA = \sqrt{(2 - 4)^2 + (0 - 5)^2} = \sqrt{29}$$

karena semua panjang sisinya berbeda, sehingga segitiga tersebut bukan segitiga sama kaki maupun segitiga sama sisi.

$$AB^2 = 13$$

$$BC^2 = 34$$

$$CA^2 = 29$$

Kemudian, periksa apakah salah satu jumlah dua sisi kuadrat sama dengan sisi ketiga:

$$13 + 29 = 42 \neq 34$$

$$13 + 34 = 47 \neq 29$$

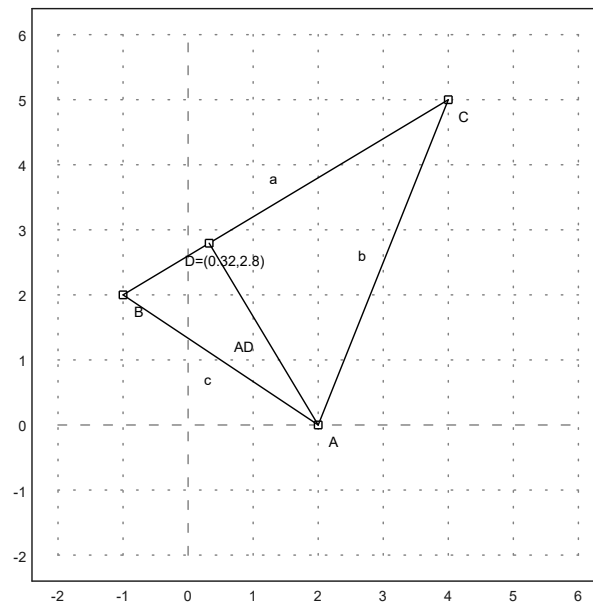
$$29 + 34 = 63 \neq 13$$

Nah, karena tidak ada yang cocok maka bukan segitiga siku-siku.
Jadi segitiga tersebut adalah segitiga sembarang.

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

```
[-3, 5, 13]
```

```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A  
>D=lineIntersection(h,lineThrough(B,C)); // D adalah titik potong h dan BC  
>plotPoint(D,value=1); // koordinat D ditampilkan  
>aspect(1); plotSegment(A,D): // tampilkan semua gambar hasil plot...()
```



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

9.5

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

9.5

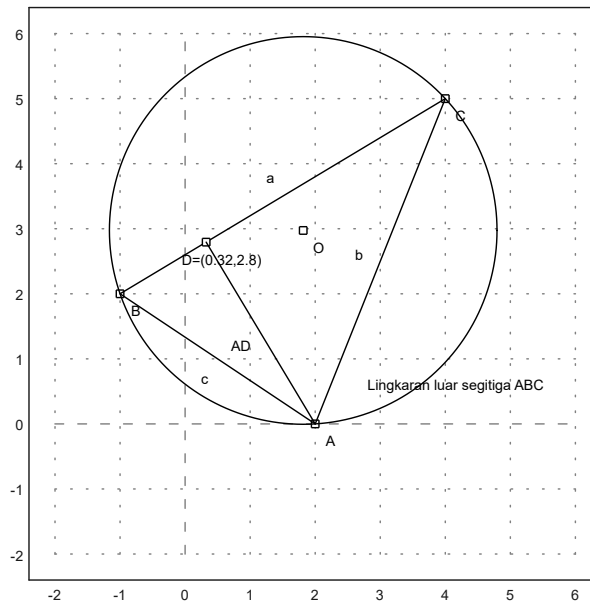
```
>distance(A,D)*distance(B,C)/2
```

9.5

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

37°14'5.40''

```
>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");
```



>O, R

```
[1.81579, 2.97368]
2.97938438305
```

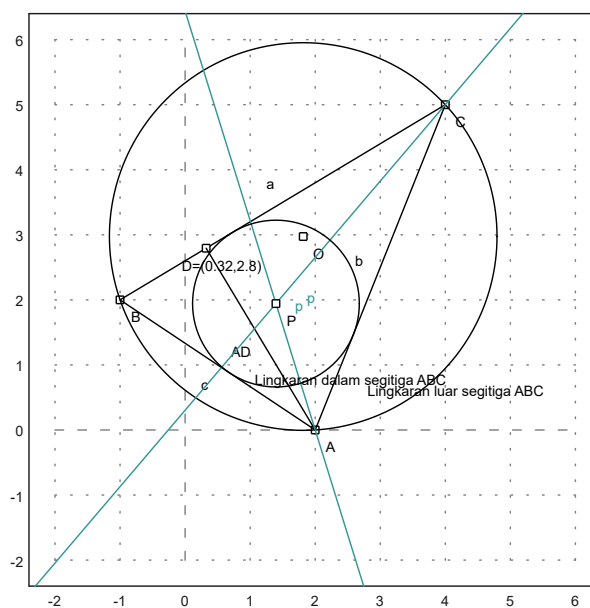
```
>l=angleBisector(A,C,B); // garis bagi <ACB
>g=angleBisector(C,A,B); // garis bagi <CAB
>P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
```

```
[1.39653, 1.94297]
```

```
>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
```

```
1.28190700459
```

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



Dari hasil tersebut berlaku hubungan:

$$[\Delta = r \cdot s = \frac{abc}{4R}]$$

yang menunjukkan bahwa luas segitiga bergantung langsung pada jari-jari lingkaran dalam dan

berbanding terbalik dengan jari-jari lingkaran luar.

Selain itu, luas kedua lingkaran tidak sama dengan luas segitiga, namun berkaitan secara proporsional melalui nilai (r) dan (R). Hasil juga memenuhi teorema Euler ($R > 2r$), yang menandakan segitiga tersebut adalah segitiga sembarang.

Jadi, dapat disimpulkan bahwa luas lingkaran dalam dan lingkaran luar memiliki hubungan tidak langsung dengan luas segitiga, tetapi saling terikat melalui jari-jari (r), (R), dan panjang sisi-sisi segitiga ABC.

Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri secara eksak dan simbolik menggunakan Maxima.

Berkas geometry.e menyediakan fungsi yang sama (dan lebih) 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-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 memperoleh persamaan suatu 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)
```

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

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

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

$$(x_1 - 1)y - xy_1 = -y_1$$

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

$$[2, 1, 2]$$

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

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

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

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

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

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

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

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

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

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

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
```

[0.86038, 0.86038]

Garis dan Lingkaran yang Berpotongan

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

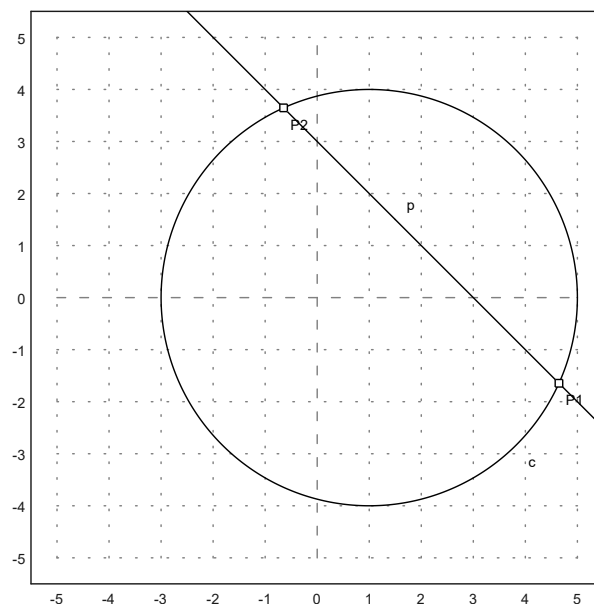
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

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

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

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

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



Sama halnya di Maxima.

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

[1, 0, 4]

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


[1, 1, 3]

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

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

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

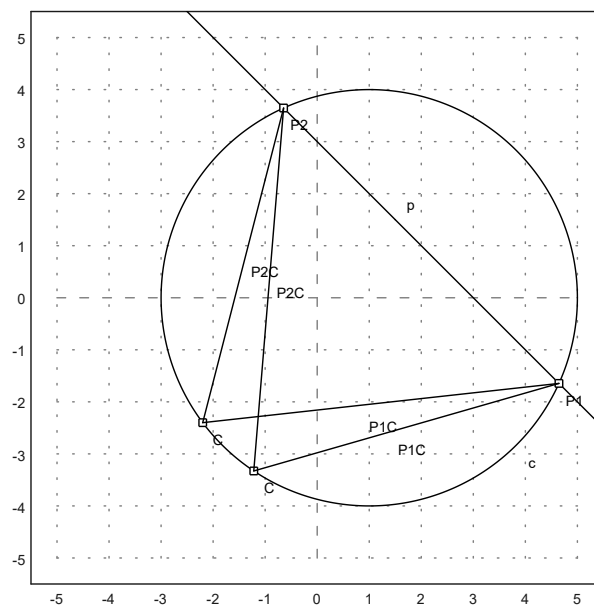
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);  
>degprint(computeAngle(P1,C,P2))
```

69°17'42.68''

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

69°17'42.68''

```
>insimg;
```

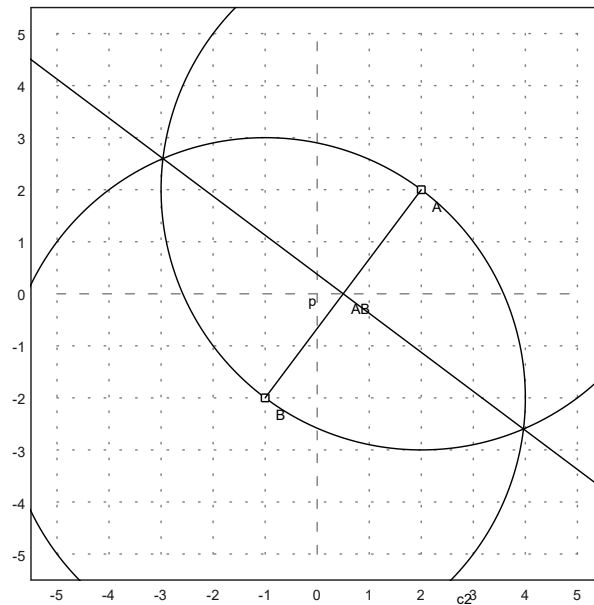


Garis Sumbu

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

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

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



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 perpotongan cukup rumit. Tetapi kita dapat menyederhanakannya jika kita menyelesaikan untuk y.

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

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

Ini memang sama dengan garis tegak lurus tengah, yang dihitung dengan cara yang sama sekali 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

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

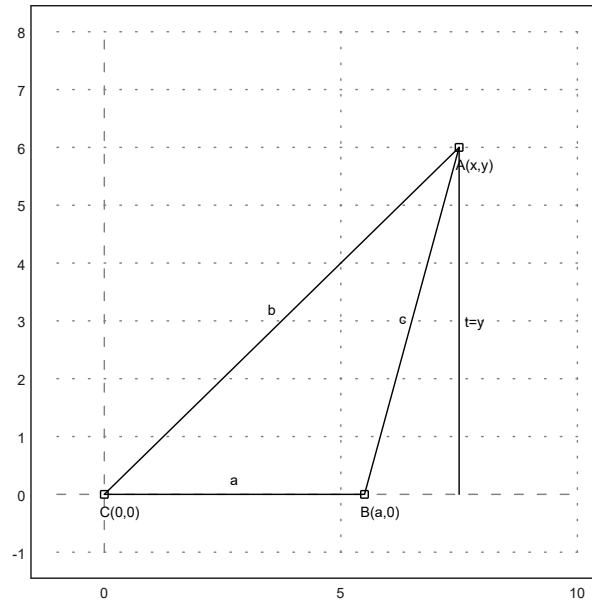
Nilai y didapat dengan menyelesaikan sistem persamaan:

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

```
>load geometry
```

Numerical and symbolic geometry.

```
>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])
```

[]

Ambil solusi y.

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

Kita memperoleh rumus Heron.

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

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

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

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

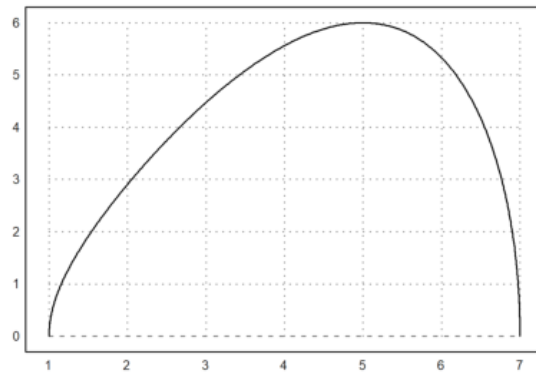
Tentu saja, setiap segitiga siku-siku merupakan kasus yang sudah dikenal dengan baik.

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

6

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 dengan panjang sisi 3, 4, x (1<= x <=7)
```



Kasus umum juga berhasil.

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

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

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

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

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

Dan buat fungsi-fungsi dari ini.

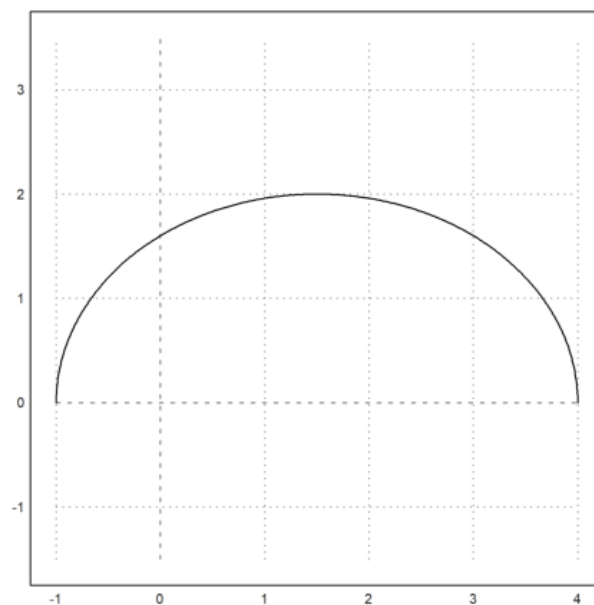
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

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

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

Sekarang kita dapat menggambar himpunan tersebut. Sisi b bervariasi dari 1 sampai 4. Diketahui bahwa kita mendapatkan sebuah elips.

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



Kita dapat memeriksa persamaan umum untuk elips ini, yaitu

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

di mana $((x_m, y_m))$ adalah pusatnya, dan (u) serta (v) adalah sumbu setengahnya.

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

1

Kita melihat bahwa tinggi dan dengan demikian luas segitiga adalah maksimum untuk $x=0$. Dengan demikian luas segitiga dengan $a+b+c=d$ adalah maksimum jika segitiga itu sama sisi. Kita ingin menurunkan ini secara analitis.

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

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

Kita memperoleh beberapa minima, yang termasuk segitiga dengan salah satu sisi 0, dan solusi $a=b=c=d/3$.

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

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

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

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

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

Kita dapat membuat plot dari situasi tersebut.

Pertama, tetapkan titik-titiknya di Maxima.

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

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

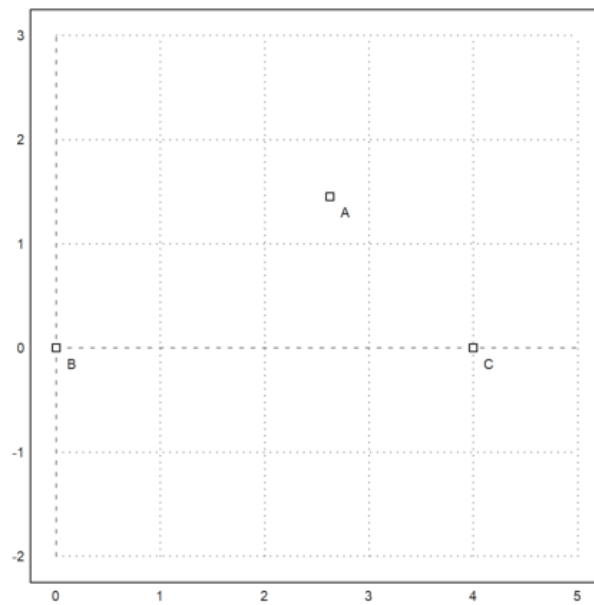
```
>B &= [0,0]; $B, C &= [a,0]; $C
```

$$[a, 0]$$

$$[a, 0]$$

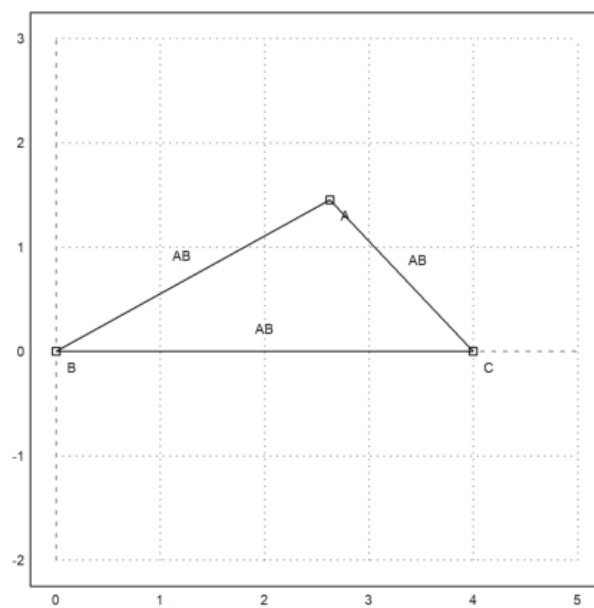
Kemudian tetapkan jangkauan plot, dan gambarkan titik-titiknya.

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



Gambarkan segmen-segmennya.

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



Hitung garis tegak lurus tengah di Maxima.

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

Dan titik pusat lingkaran keliling.

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

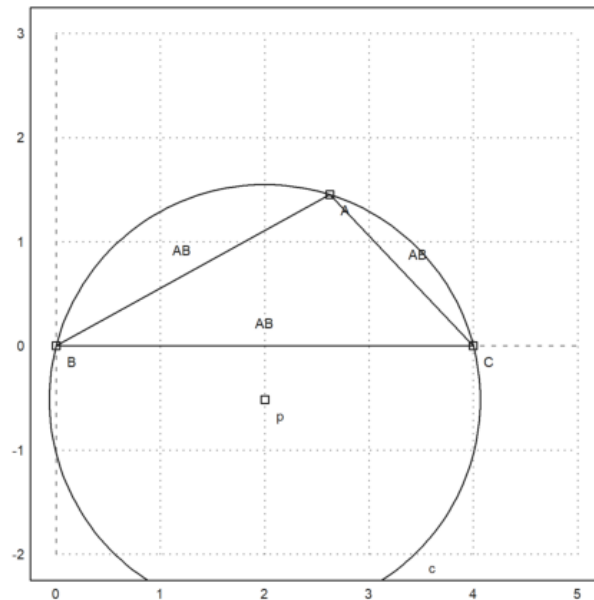
Kita memperoleh rumus untuk jari-jari lingkaran keliling.

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

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

Mari kita tambahkan ini ke plot.

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



Dengan menggunakan geometri, kita menurunkan rumus sederhana

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

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

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

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

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari setiap segitiga yang tidak sama sisi. Garis ini merupakan garis pusat segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga tersebut, termasuk ortosenter, pusat lingkaran keliling, titik berat, titik Exeter, dan pusat lingkaran sembilan titik dari segitiga itu.

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

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

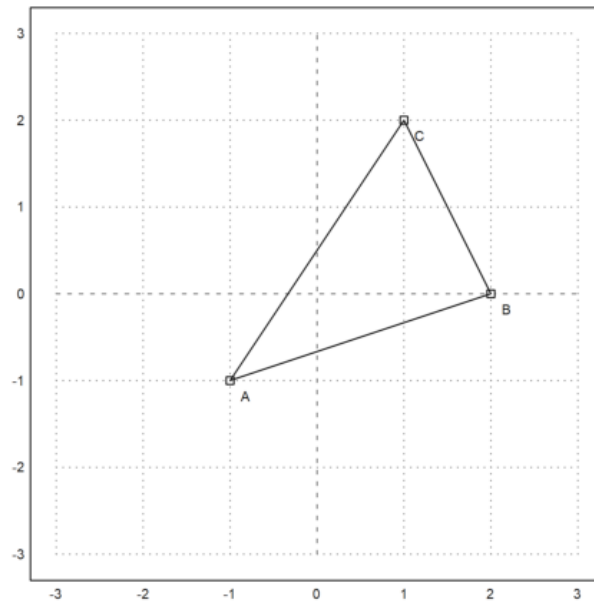
```
>A:=[-1,-1]; B:=[2,0]; C:=[1,2];
```

Untuk menggambar objek geometri, kita menyiapkan area plot dan menambahkan titik-titik ke dalamnya. Semua plot dari objek geometri ditambahkan ke plot yang sedang aktif.

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

Kita juga dapat menambahkan sisi-sisi segitiga.

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



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

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

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

Kita dapat menghitung koefisien-koefisien dari sisi c.

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

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

Dan juga mendapatkan rumus untuk garis ini.

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

$$3y - x = -2$$

Untuk bentuk Hesse, kita perlu menentukan sebuah titik sedemikian rupa sehingga titik tersebut berada di sisi positif dari bentuk Hesse. Dengan memasukkan titik tersebut, kita memperoleh jarak positif terhadap 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 keliling segitiga ABC.

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

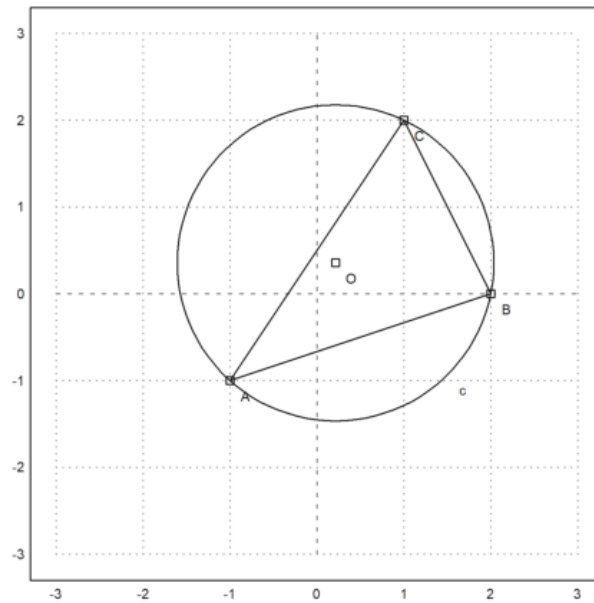
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

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

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

Gambarkan lingkaran dan titik pusatnya. Cu dan U bersifat simbolik. Kita mengevaluasi ekspresi-ekspresi ini untuk Euler.

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

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

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

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

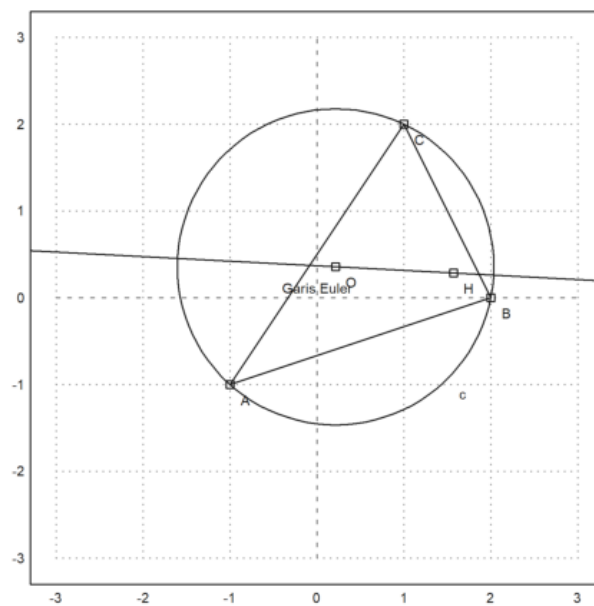
Sekarang kita dapat menghitung garis Euler dari segitiga tersebut.

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

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

Tambahkan itu ke plot kita.

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

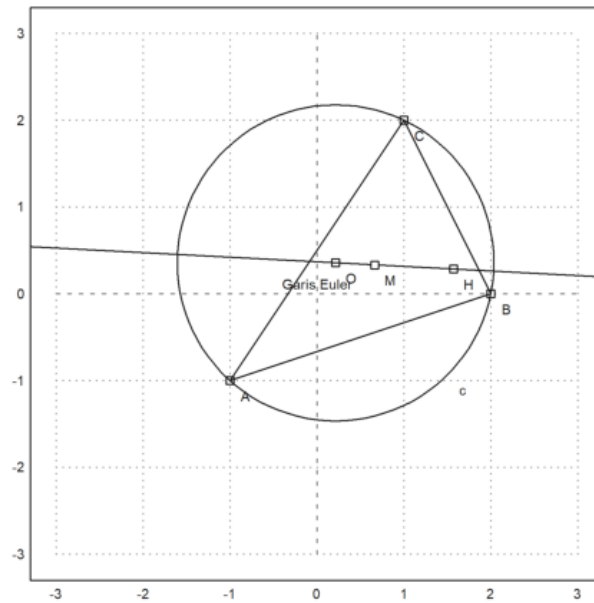


Titik berat seharusnya berada pada garis ini.

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

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

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



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

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

2

Fungsi-fungsinya 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 sederhana.

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

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

Mari kita hitung juga ekspresi untuk jari-jari lingkaran dalam.

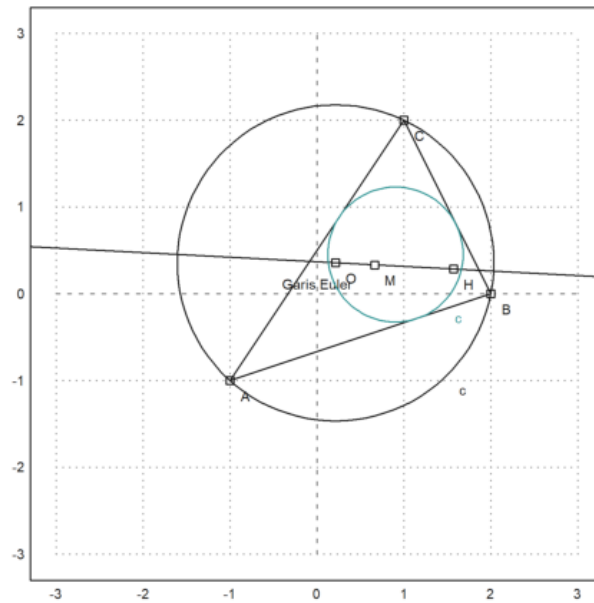
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
```

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

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

Mari kita tambahkan ini ke plot.

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



Parabola

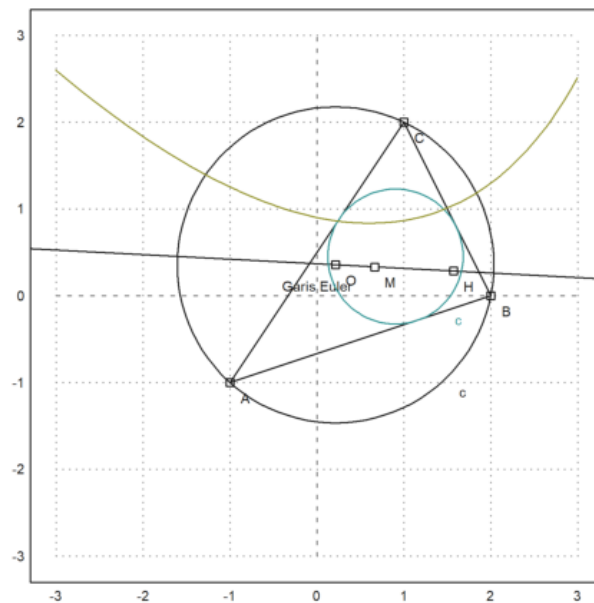
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

```
>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 penyelesaian bawaan Maxima hanya dapat menemukan solusinya jika kita mengkuadratkan persamaannya. Akibatnya, kita mendapatkan solusi palsu.

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

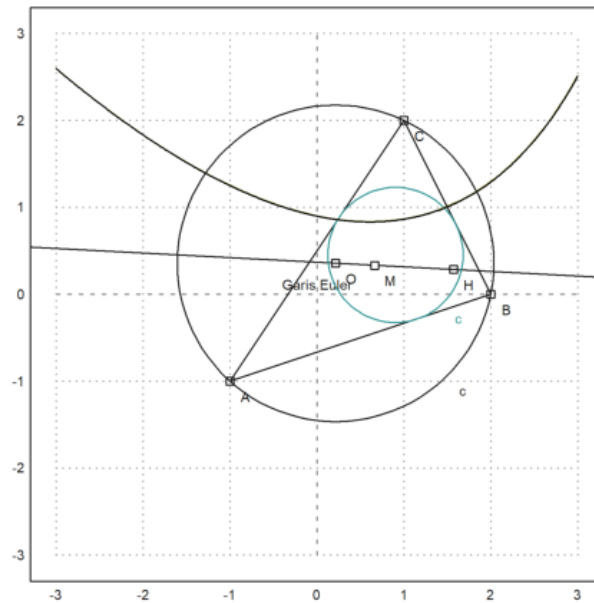
$$\begin{aligned} [y &= -3x - \sqrt{70} \sqrt{9 - 2x} + 26, \\ y &= -3x + \sqrt{70} \sqrt{9 - 2x} + 26] \end{aligned}$$

Solusi pertama adalah

$akar_1$

Menambahkan solusi pertama ke plot menunjukkan bahwa itu memang lintasan yang kita cari. Teori menyatakan bahwa itu adalah sebuah 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
```

2.135605779339061

2.135605779339061

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

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

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

2.135605779339061

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

Bagian ini terinspirasi oleh ceramah N. J. Wildberger. Dalam bukunya *Divine Proportions*, Wildberger mengusulkan untuk menggantikan konsep klasik jarak dan sudut dengan *quadrance* dan *spread*. Dengan menggunakan konsep-konsep ini, memang dimungkinkan untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap bersifat "rasional".

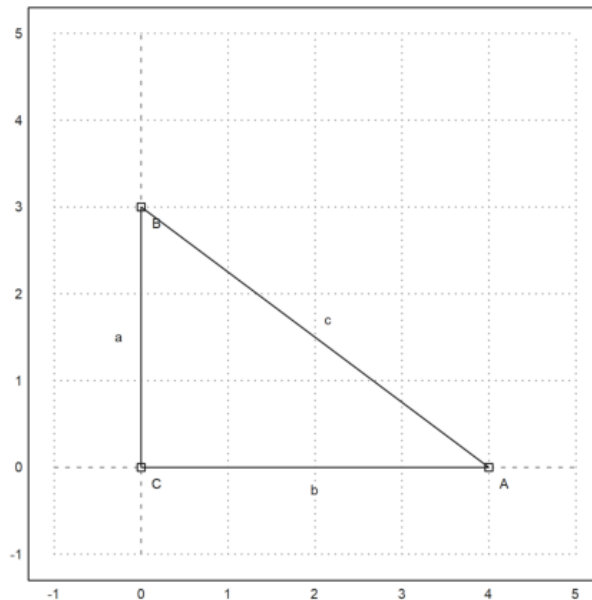
Dalam bagian berikut, saya memperkenalkan konsep-konsep tersebut dan menyelesaikan beberapa masalah. Di sini saya menggunakan perhitungan simbolik Maxima, yang sebenarnya menyembunyikan keunggulan utama trigonometri rasional — yaitu bahwa perhitungannya dapat dilakukan hanya dengan pena dan kertas. Kamu dipersilakan untuk memeriksa hasilnya tanpa komputer.

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

```
>load geometry;
```

Sebagai pengenalan pertama, kita menggunakan segitiga siku-siku dengan perbandingan Mesir yang terkenal, yaitu 3, 4, dan 5. Perintah-perintah berikut adalah perintah Euler untuk menggambar geometri bidang yang terdapat dalam berkas Euler "geometry.e".

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



Tentu saja,

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

di mana (w_a) adalah sudut di A. Cara biasa untuk menghitung sudut ini adalah dengan mengambil invers dari fungsi sinus. Hasilnya adalah sudut yang sulit disederhanakan, yang hanya dapat ditampilkan secara perkiraan.

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

```
36°52'11.63''
```

Trigonometri rasional berusaha untuk menghindari hal ini.

Konsep pertama dalam trigonometri rasional adalah *quadrance*, yang menggantikan jarak. Sebenarnya, *quadrance* hanyalah jarak yang dikuadratkan. Dalam hal ini, a, b, dan c menyatakan *quadrance* dari sisi-sisi segitiga.

Teorema Pythagoras kemudian menjadi sederhana: ($a + b = c$.)

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

```
25 = 25
```

Konsep kedua dalam trigonometri rasional adalah spread. Spread mengukur besar bukaan antara dua garis. Nilainya 0 jika kedua garis sejajar, dan 1 jika kedua garis saling tegak lurus. Nilai ini merupakan kuadrat dari sinus sudut antara dua garis tersebut.

Spread 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 quadrance dari sembarang segitiga siku-siku dengan salah satu sudut di A.

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

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

Tentu saja, ini lebih mudah dihitung daripada sudut. Namun, kita kehilangan sifat bahwa sudut dapat dijumlahkan dengan mudah.

Tentu saja, kita dapat mengonversi nilai pendekatan untuk sudut (w_a) menjadi spread, dan menampilkannya dalam bentuk pecahan.

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

```
9/25
```

Hukum kosinus dalam trigonometri klasik diterjemahkan ke dalam "hukum silang" (cross law) berikut:

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

Di sini (a), (b), dan (c) adalah quadrance dari sisi-sisi segitiga, dan (s_a) adalah spread pada sudut A. Sisi (a) seperti biasa berhadapan dengan sudut A.

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

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

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

Dalam kasus kita, kita memperoleh

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

$$1024 = 1024$$

Mari gunakan hukum silang ini untuk menemukan spread di A. Untuk melakukan ini, kita menghasilkan hukum silang untuk quadrance a, b, dan c, dan menyelesaikannya untuk spread tak dikenal sa.

Kamu bisa melakukan ini dengan tangan dengan mudah, 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 juga dapat menyelesaikan ini untuk a,b,c umum. Hasilnya adalah sebuah rumus yang menghitung spread suatu sudut segitiga dengan diberikan quadrance ketiga sisinya.

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

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

Kita dapat membuat sebuah fungsi dari hasil tersebut. Fungsi seperti ini sebenarnya sudah didefinisikan dalam berkas geometry.e milik Euler.

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

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

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

$$(a, a, \frac{4a}{7})$$

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

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

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

Ini adalah sudut dalam derajat.

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

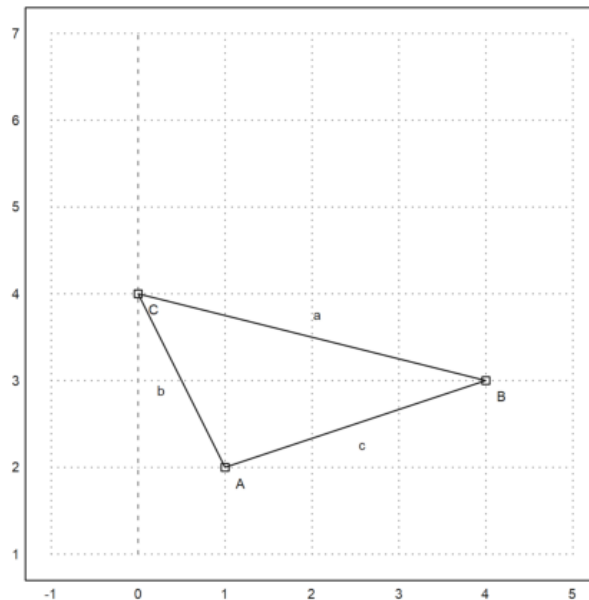
$$67^{\circ}47'32.44''$$

Contoh Lain

Sekarang, mari kita coba contoh yang lebih lanjut.

Kita tetapkan 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 Teorema Pythagoras, mudah untuk menghitung jarak antara dua titik. Saya terlebih dahulu menggunakan fungsi distance dari berkas geometry milik Euler. 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) , maka segitiga tersebut bukan segitiga siku-siku.

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

$$17$$

$$5$$

$$17$$

Pertama, mari kita hitung sudut tradisionalanya. Fungsi `*computeAngle*` menggunakan metode biasa yang didasarkan pada hasil kali titik (dot product) dari dua vektor. Hasilnya berupa pendekatan bilangan pecahan desimal.

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

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

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

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

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

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

$$32.4711922908$$

Dengan pensil dan kertas, kita dapat melakukan hal yang sama dengan hukum silang. Kita memasukkan quadrance a, b, dan c ke dalam hukum silang dan menyelesaikannya untuk x.

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

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

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

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

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

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

Maxima mendapatkan hasil yang sama dengan menggunakan trigonometri biasa, jika kita memaksanya. Program tersebut dapat menyederhanakan bentuk ($\sin(\arccos(\dots))$) menjadi hasil pecahan. Sebagian besar siswa tidak akan dapat melakukan hal ini secara manual.

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

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

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

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

berdasarkan definisi.

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

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

Gambar berikut dibuat dengan program geometri C.a.R., yang dapat menggambar quadrance dan spread.

image: (20) Rational_Geometry_CaR.png

Menurut definisi, panjang (ha) adalah akar kuadrat dari quadrance-nya.

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

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

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

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

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

Rumus determinan yang 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 spread 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.

Harus diakui, ini sulit dilakukan dengan pensil dan kertas.

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

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

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

Aturan Triple Spread

Kelemahan dari spread adalah bahwa nilainya tidak dapat dijumlahkan secara sederhana seperti sudut.

Namun, tiga spread dari suatu segitiga memenuhi aturan "triple spread" berikut.

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

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

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

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

Karena spread dari

$$(\alpha, \pi - \alpha)$$

adalah sama, maka aturan triple spread juga berlaku jika

$$(\alpha + \beta = \gamma)$$

Selain itu, karena spread dari sudut negatif sama, aturan triple spread juga tetap berlaku jika

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

Sebagai contoh, kita dapat menghitung spread untuk sudut 60° . Nilainya adalah $(\frac{3}{4})$. Namun, persamaan tersebut memiliki solusi kedua, yaitu ketika semua *spread* bernilai 0.

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

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

Spread dari sudut 90° jelas bernilai 1. Jika dua sudut berjumlah 90° , maka spread-nya memenuhi persamaan triple spread dengan (a, b, 1). Dari perhitungan berikut, kita memperoleh bahwa (a + b = 1.)

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

$$[x = 1 - y]$$

$$[x = 1 - y]$$

Karena spread dari $(180^\circ - t)$ sama dengan spread dari (t), maka rumus triple spread juga berlaku jika satu sudut merupakan jumlah atau selisih dari dua sudut lainnya.

Dengan demikian, kita dapat menemukan spread dari sudut yang digandakan. Perhatikan bahwa kembali ada dua solusi. Kita jadikan ini sebuah fungsi.

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

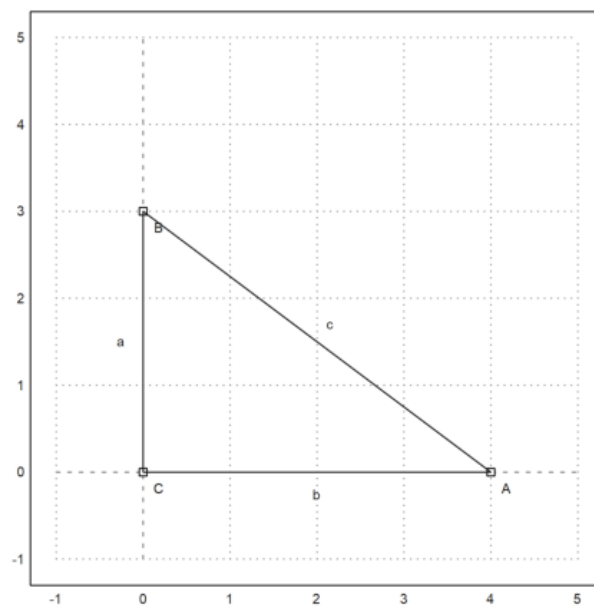
$$[x = 4a - 4a^2, x = 0]$$

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

Pembagi Sudut (Angle Bisectors)

Ini adalah situasi yang sudah kita kenal.

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



Mari kita hitung panjang garis bagi sudut di A. Tetapi kita ingin menyelesaikannya untuk a,b,c umum.

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

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

Masalah dengan rumus ini muncul lagi. Ia memiliki 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
```

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

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

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

Mari kita periksa untuk segitiga Mesir.

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

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

Kita dapat menampilkan sudutnya di Euler setelah mengubah spread menjadi radian.

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

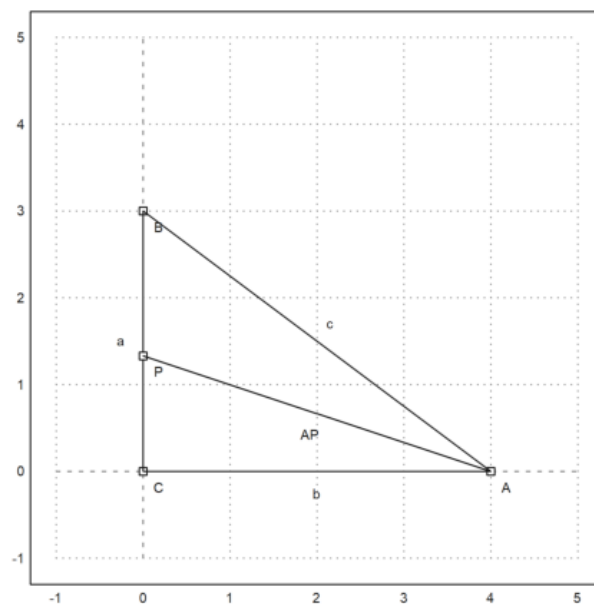
18°26'5.82''

Titik P adalah titik 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-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 pada setiap segitiga. Jika dikuadratkan, itu diterjemahkan menjadi yang disebut "spread law"

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

di mana a,b,c menyatakan qudrances.

Karena spread CPA adalah 1-sa2, kita memperoleh darinya bisa/1=b/(1-sa2) dan dapat menghitung bisa (quadrance 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 Mesir kita.

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

```
4.21637021356
4.21637021356
```

Kita juga dapat menghitung P menggunakan rumus spread.

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

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

Nilainya sama dengan yang kita peroleh menggunakan rumus trigonometri.

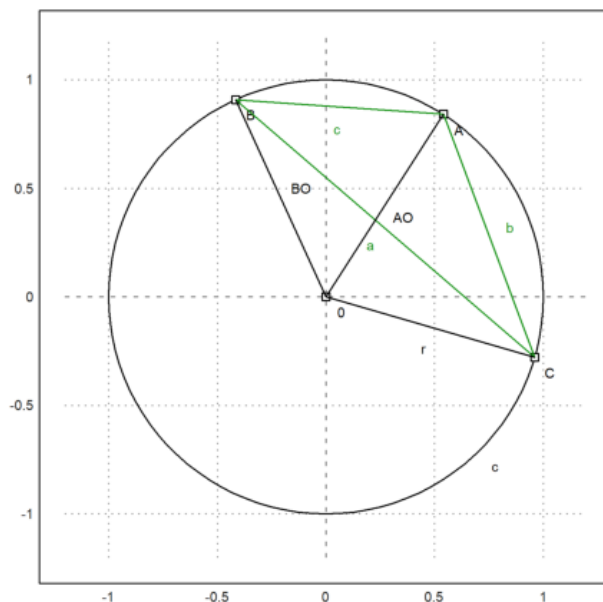
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

```
1.33333333333
```

Sudut Tali Busur (The Chord Angle)

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-sudut di pusat O terhadap r. Dengan demikian kita mendapatkan rumus untuk kuadrat jari-jari pericircle dalam hal quadrance sisi-sisi.

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

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

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

Kita dapat membuatnya menjadi sebuah fungsi Euler.

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

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

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

Jari-jari memang 1.

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

1

Faktanya, bahwa sebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut tali busur.

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

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

Sebenarnya sebaran adalah $b/(4r)$, dan kita melihat bahwa sudut tali busur dari tali b adalah setengah dari sudut pusat.

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

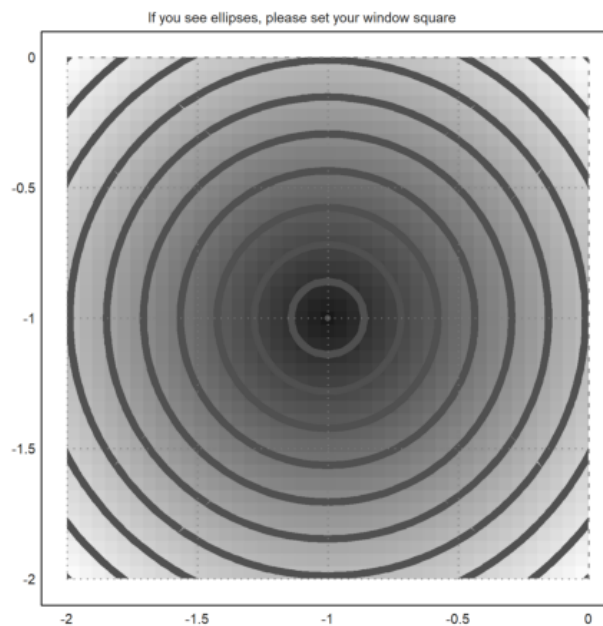
0

Contoh 6: Jarak Minimal pada Bidang

Catatan pendahuluan

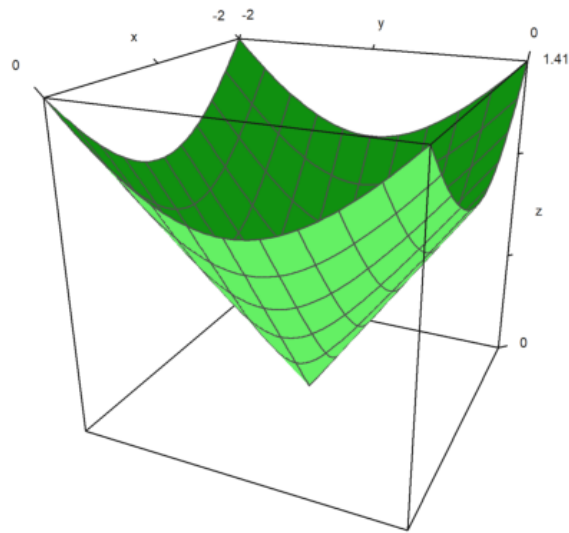
Fungsi yang, untuk suatu titik M pada bidang, memberikan jarak AM antara titik tetap A dan M, memiliki garis level yang cukup sederhana: lingkaran berpusat di A.

```
>&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 dari sebuah kerucut:

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

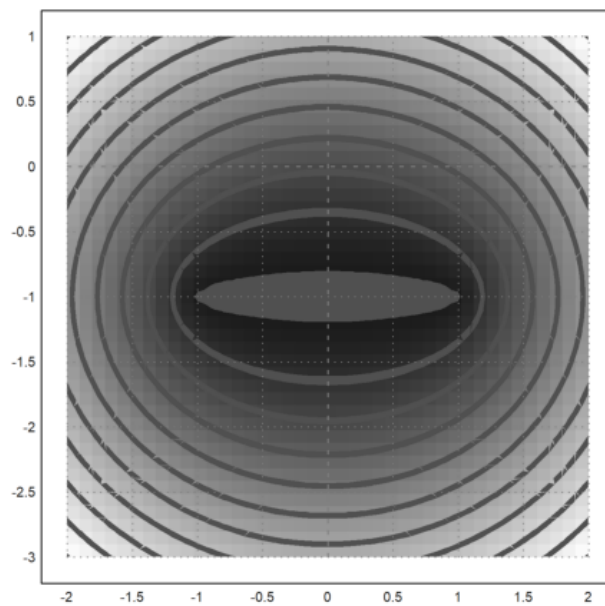


Tentu saja minimum 0 dicapai di A.

Dua titik

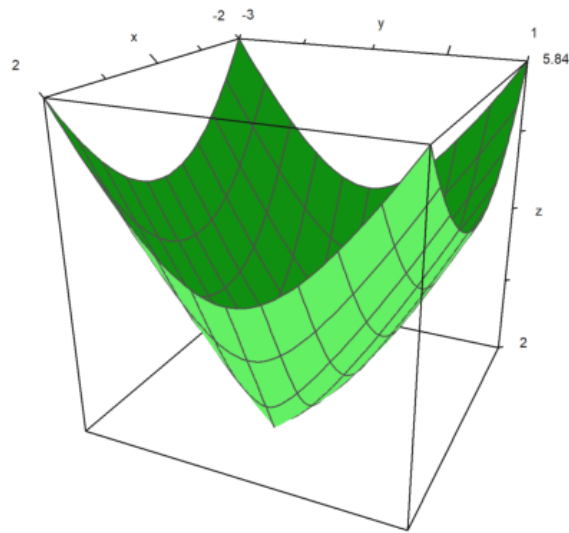
Sekarang kita melihat fungsi $MA+MB$ di mana A dan B adalah dua titik (tetap). Ini adalah "fakta yang sudah dikenal" bahwa kurva levelnya adalah elips, titik fokusnya adalah A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

```
>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):
```



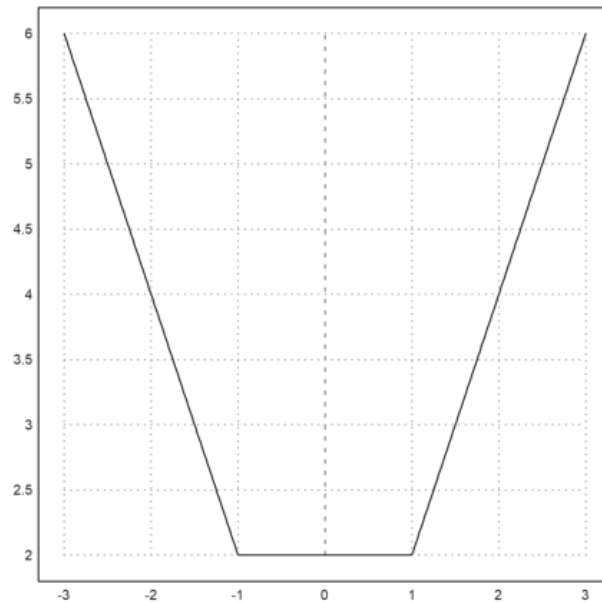
Grafiknya lebih menarik:

```
>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):
```



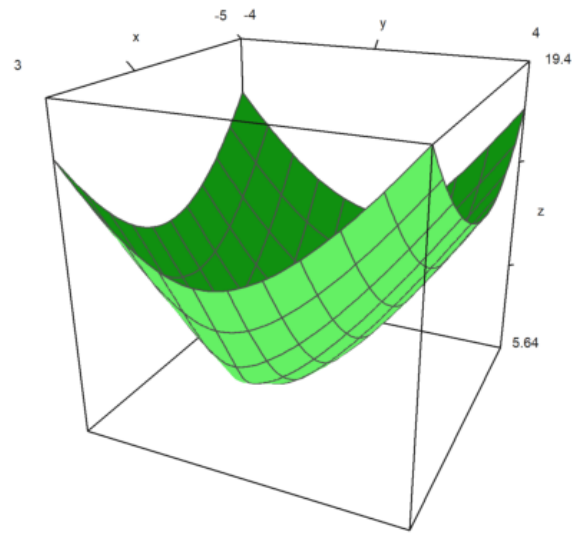
Tiga titik

Sekarang keadaan kurang sederhana: agak kurang dikenal bahwa $MA+MB+MC$ mencapai nilai minimumnya di satu titik pada bidang tetapi untuk menentukannya lebih rumit:

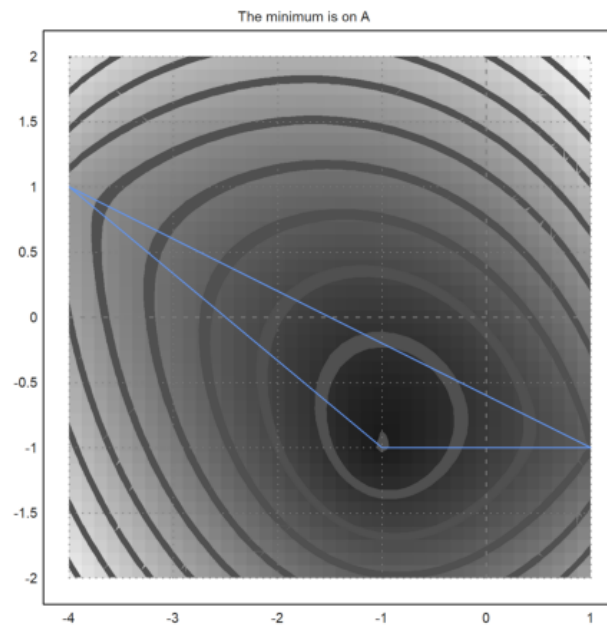
1. Jika salah satu sudut segitiga ABC lebih dari 120° (misalnya di A), maka nilai minimum dicapai di titik itu sendiri (misalnya $AB+AC$).

Contoh:

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

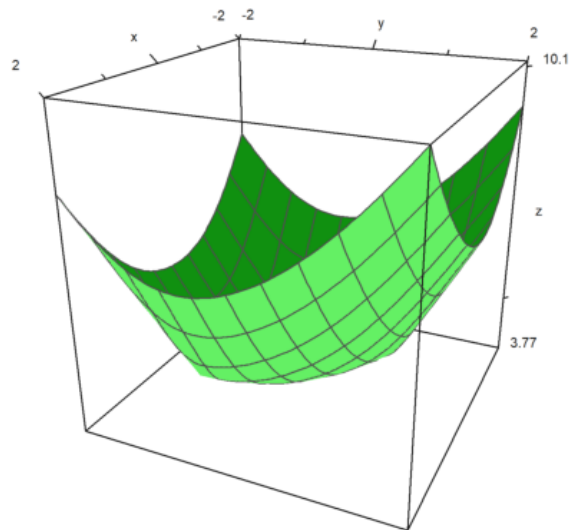


```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

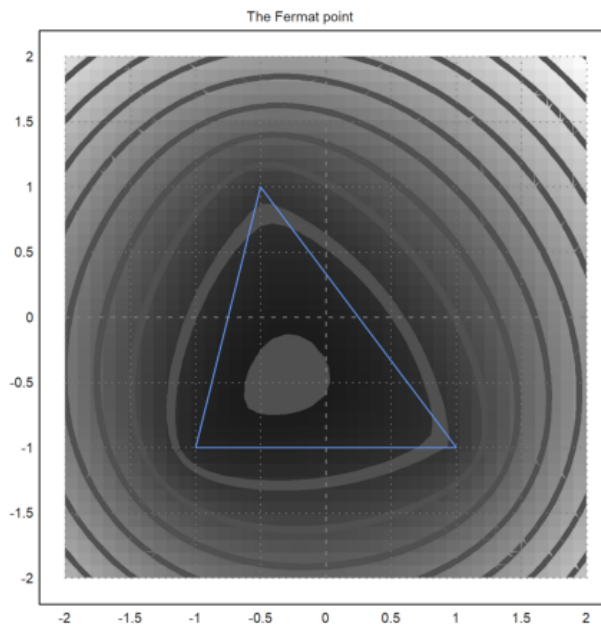


2. Tetapi jika semua sudut segitiga ABC kurang dari 120° , maka minimumnya berada pada suatu titik F di dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (masing-masing 120°):

```
>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;
```



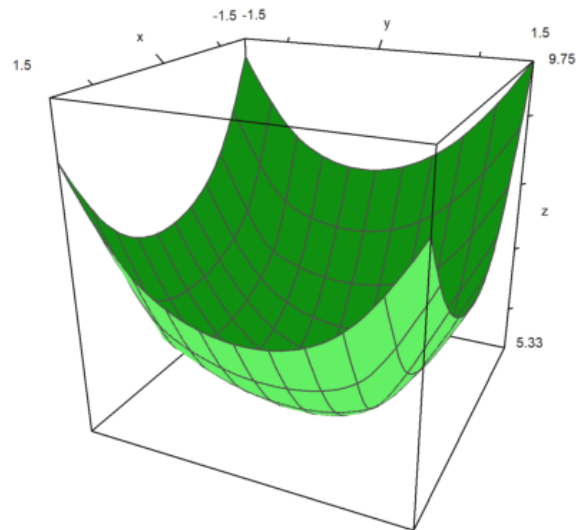
Ini adalah kegiatan yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu sebuah perangkat lunak yang ditulis dalam Java yang memiliki perintah "contour lines"...

Semua hal di atas ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; ia menulis surat kepada para amatir lain seperti imam Marin Mersenne dan Blaise Pascal yang bekerja pada pajak penghasilan. Jadi titik unik F sedemikian sehingga $FA+FB+FC$ minimal, disebut titik Fermat dari segitiga. Tetapi tampaknya beberapa tahun sebelumnya, orang Italia Torricelli telah menemukan titik ini sebelum Fermat melakukannya! Bagaimanapun tradisinya adalah memberi notasi titik ini F...

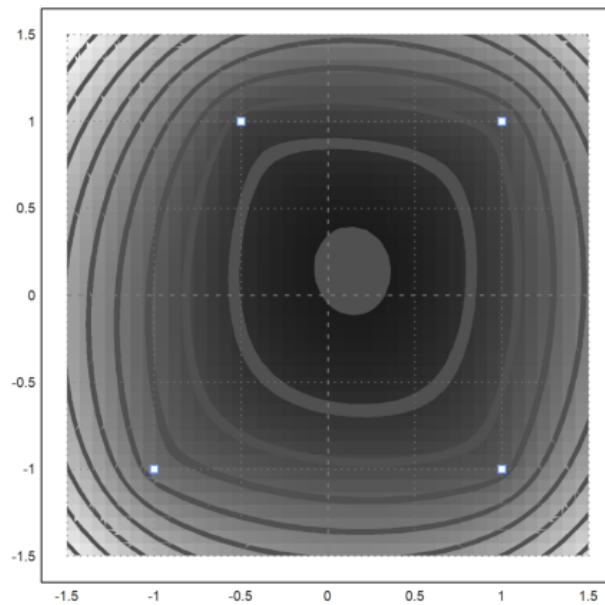
Empat titik

Langkah berikutnya adalah menambahkan titik ke-4 D dan mencoba meminimalkan $MA+MB+MC+MD$; misalkan Anda adalah operator TV kabel dan ingin mencari di bidang mana Anda harus menempatkan antenna sehingga Anda dapat memberi 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 terdapat sebuah minimum dan itu tidak dicapai di salah satu titik A, B, C, maupun D:

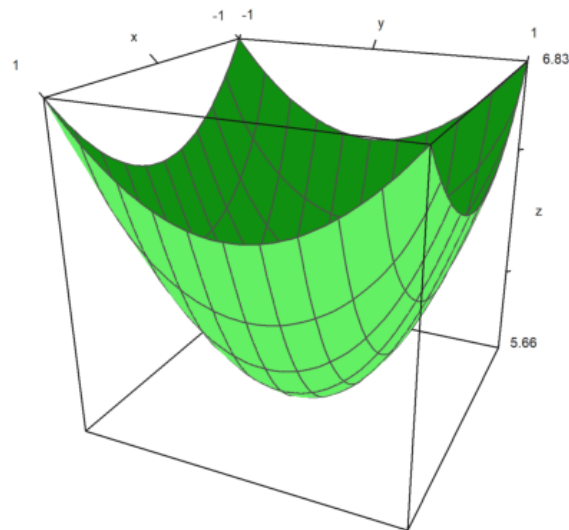
```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```

```
[0.142858, 0.142857]
```

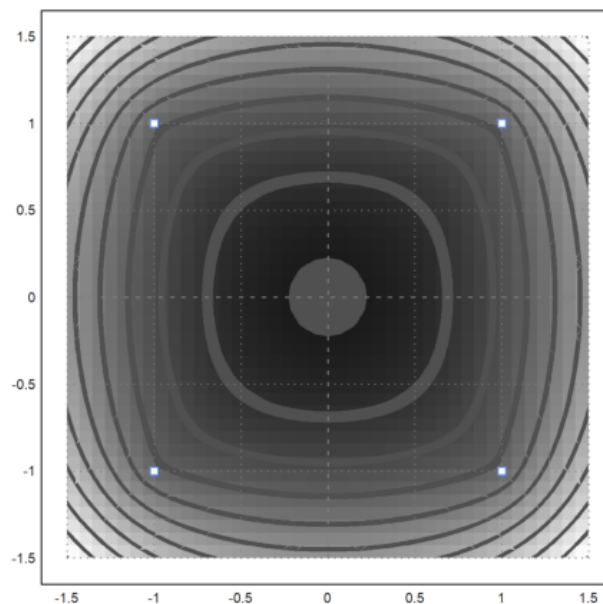
Tampaknya dalam kasus ini, koordinat titik optimal bersifat rasional atau hampir rasional...

Sekarang ABCD adalah sebuah persegi, kita mengharapkan bahwa titik optimalnya 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;
```



Contoh 7: Bola Dandelin dengan Povray

Anda dapat menjalankan demonstrasi ini, jika Anda telah menginstal Povray, dan pvengine.exe ada di path 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, dua garis yang membentuk kerucut.

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

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

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

$[-a, -1, 0]$

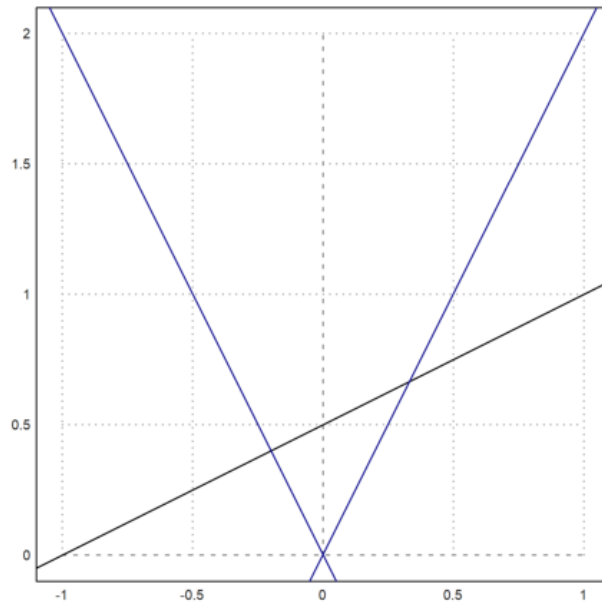
Kemudian, garis ketiga.

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

$[-1, 2, 1]$

Kita plot semua sejauh ini.

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



Now we take a general point on the y-axis.

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

$[0, u]$

Compute the distance to 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 temukan pusat dari dua lingkaran, di mana jaraknya sama.

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

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

Ada dua solusi.

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

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

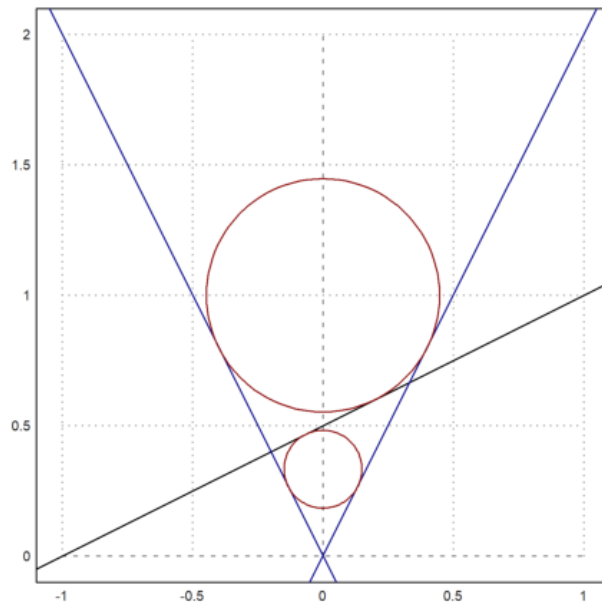
```
[0.333333, 1]
```

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

```
[0.149071, 0.447214]
```

Plot lingkaran-lingkaran ke dalam gambar.

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



Plot dengan Povray

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

Pertama, kita memuat 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 menyiapkan adegan dengan tepat.

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

Selanjutnya kita tulis dua bola tersebut 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 kerucutnya, 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, tempat 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 tempat bola menyentuh bidang. Titik-titik 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 menghitung perpotongan P1P2 dengan bidang.

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

Kita hubungkan titik-titik tersebut dengan segmen garis.

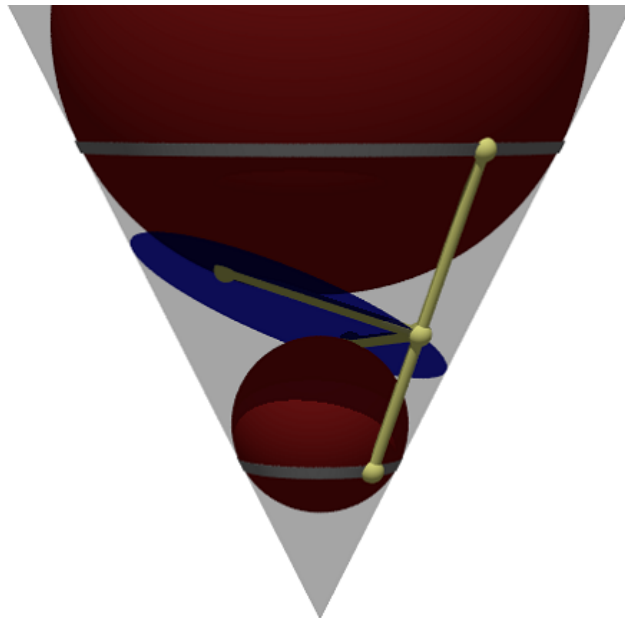
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan pita abu-abu, tempat bola menyentuh kerucut.

```
>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)));
```

Jalankan program Povray.

```
>povend();
```



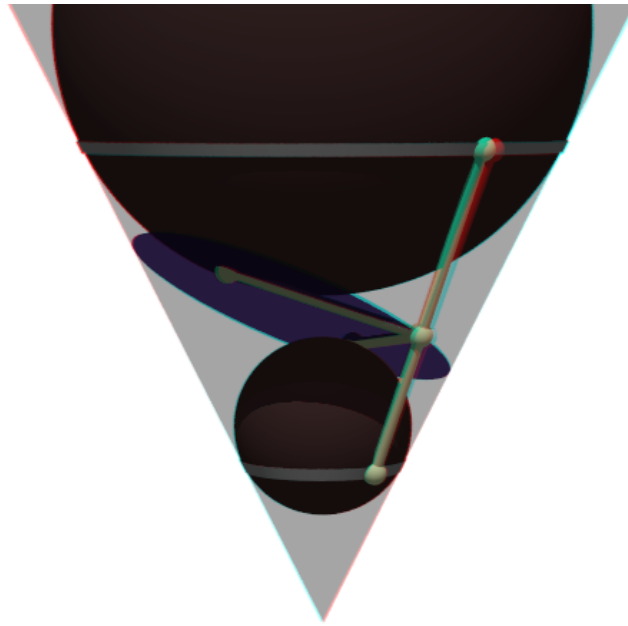
Untuk mendapatkan Anaglyph dari ini, kita perlu menempatkan semuanya ke dalam fungsi adegan. Fungsi ini akan digunakan dua kali nanti.

```
>function scene () ...
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g();
pc=povcone([0,0,0],0,[0,0,a],1,"");
vp=[gp[1],0,gp[2]]; dp=gp[3];
writeln(povplane(vp,dp,povlook(blue,0.5),pc));
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
writeln(povpoint(P1,povlook(yellow)));
P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
writeln(povpoint(P2,povlook(yellow)));
P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
writeln(povpoint(P3,povlook(yellow)));
P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
writeln(povpoint(P4,povlook(yellow)));
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1);
writeln(povpoint(P5,povlook(yellow)));
writeln(povsegment(P1,P2,povlook(yellow)));
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/sian 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 bola. Fungsi-fungsinya terdapat dalam file "spherical.e" di folder contoh. Kita perlu memuat file tersebut 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 ini adalah koordinat untuk Kampus FMIPA UNY.

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

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

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

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

```
S 7°46.467' E 110°23.050'
```

Mari kita tambahkan dua kota lagi, yaitu 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 ke yang lain pada bola ideal. Vektor ini adalah [arah, jarak] dalam radian. Untuk menghitung jarak di bumi, kita mengalikannya 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. Rutin berikut menggunakan pendekatan yang bahkan lebih baik. Pada jarak yang begitu pendek, hasilnya hampir sama.

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

```
88.0114026318 km
```

Ada fungsi untuk arah yang memperhitungkan bentuk elips bumi. Sekali lagi, kita mencetaknya dengan cara yang lebih canggih.

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

65.34°

Sudut sebuah 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, hal ini tidak akurat karena kesalahan pengurangan pada asum-pi.

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

2116.02948749 km^2

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

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

2123.64310526 km^2

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

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

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

Fungsi-fungsi ini mengasumsikan bola ideal. Hal yang sama berlaku 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'

Let us turn to a larger example, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

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

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

Menurut Google Earth, jaraknya adalah 429,66 km. Kita mendapatkan perkiraan yang baik.

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

431.565659488 km

Arah tersebut 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 arah dan jarak ke posisi awal. Hal ini terjadi karena kita tidak menghitung fungsi invers secara tepat, melainkan menggunakan pendekatan terhadap jari-jari bumi 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 saja, kita tidak dapat berlayar dengan arah yang sama dari satu tujuan ke tujuan lain jika kita ingin menempuh jalur terpendek. Bayangkan, Anda terbang ke arah timur laut mulai dari titik mana pun di bumi. Maka Anda akan berputar menuju kutub utara. Lingkaran besar tidak mengikuti arah yang konstan!

Perhitungan berikut menunjukkan bahwa kita akan sangat melenceng dari tujuan yang benar jika kita menggunakan arah yang sama selama perjalanan.

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

Sekarang kita tambahkan 10 kali sepersepuluh dari jarak, menggunakan arah menuju Monas yang kita peroleh di Tugu.

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

Hasilnya sangat melenceng.

```
>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 ke P2 bukanlah lingkaran pada lintang 30°, melainkan jalur yang lebih pendek yang dimulai 10° lebih ke utara di P1.

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

79.69°

Namun, jika kita mengikuti pembacaan kompas ini, kita akan berputar menuju kutub utara! Jadi, kita harus menyesuaikan arah kita sepanjang perjalanan. Untuk perkiraan kasar, kita menyesuaikannya setiap 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°

Jarak-jarak itu 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 kita 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 dapat memperoleh serangkaian 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 memplot bumi, dua posisi tersebut, dan posisi-posisi di antaranya.

```

>function testplot ...
useglobal;
plotearth;
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");
plotposline(v);
endfunction

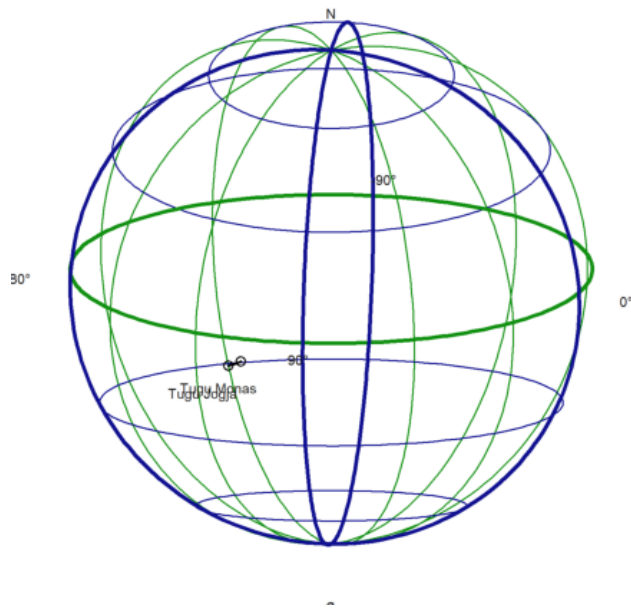
```

Sekarang plot semuanya.

```

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

```

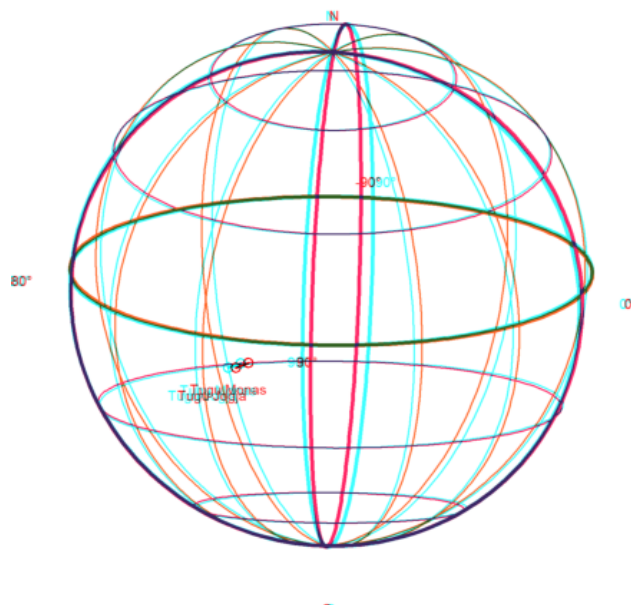


Atau gunakan plot3d untuk mendapatkan tampilan anaglifnya. Ini terlihat sangat bagus dengan kacamata merah/sian.

```

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

```



1. Gambarkan 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.

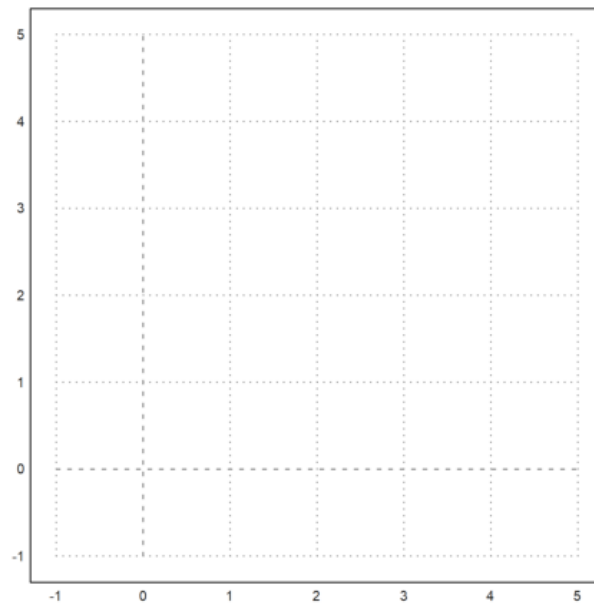
Petunjuk:

- 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.

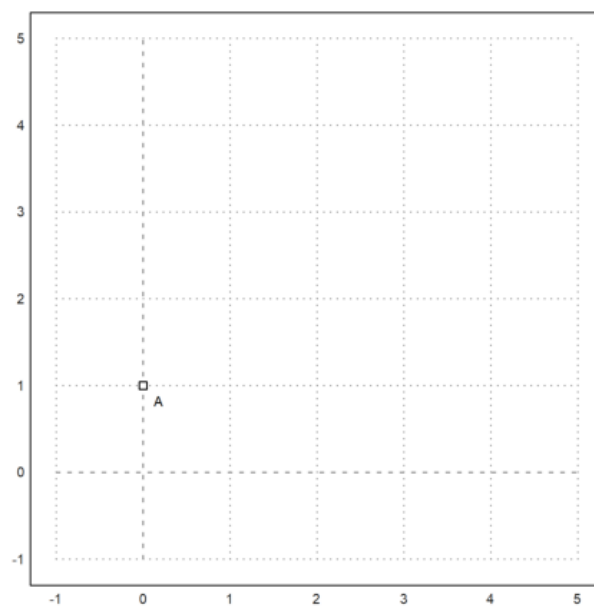
```
>load geometry
```

Numerical and symbolic geometry.

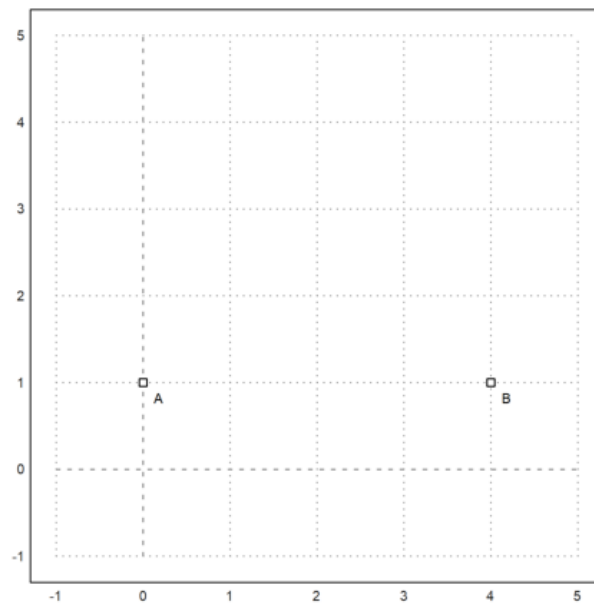
```
>setPlotRange(-1,5,-1,5):
```



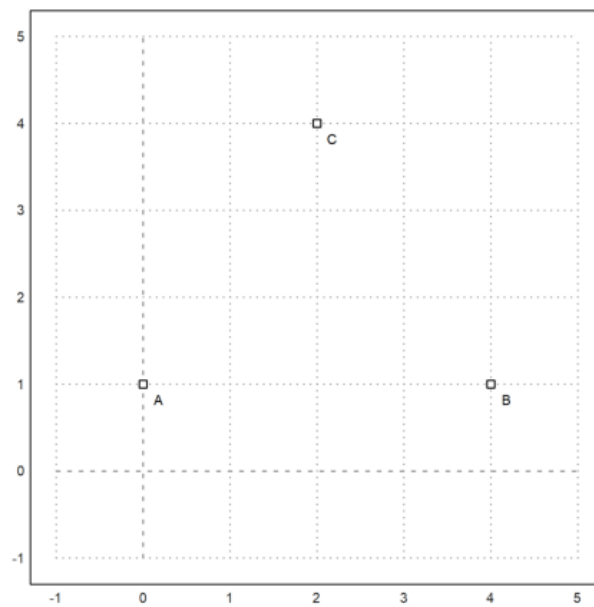
```
>A=[0,1]; plotPoint(A,"A"):
```



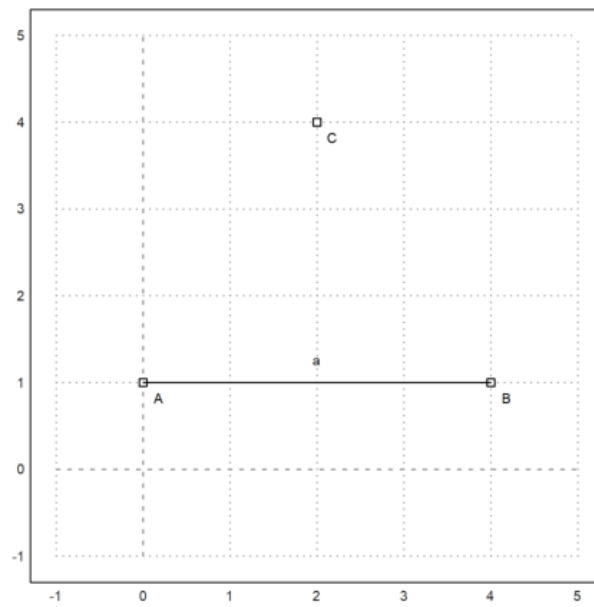
```
>B=[4,1]; plotPoint(B,"B"):
```



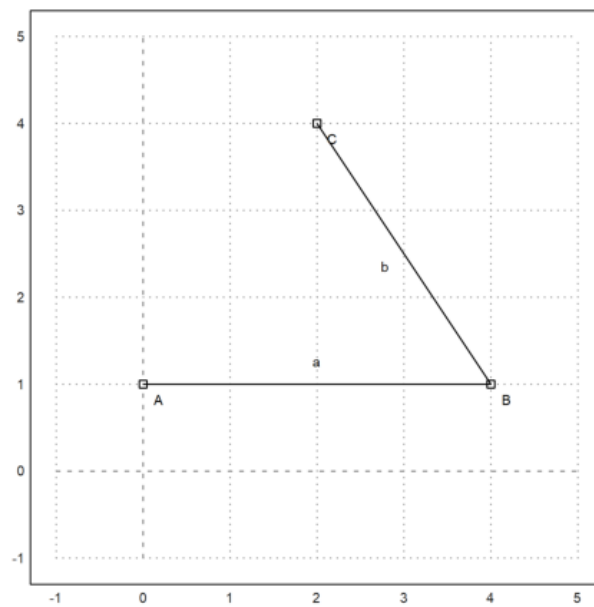
```
>C=[2,4]; plotPoint(C,"C"):
```



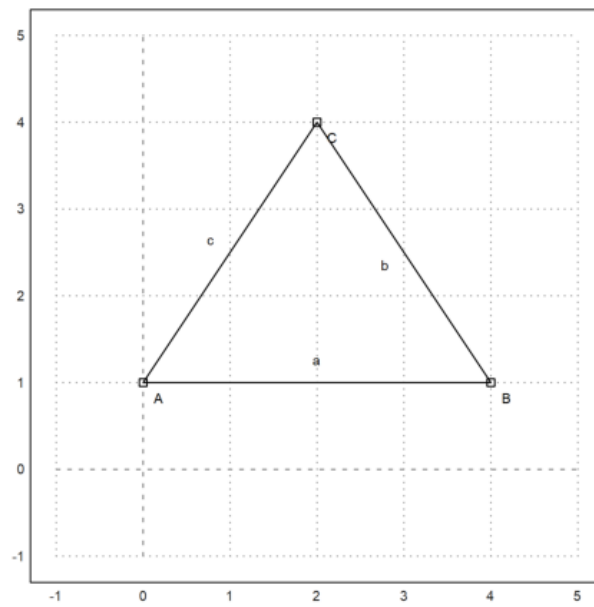
```
>plotSegment(A,B,"a"):
```



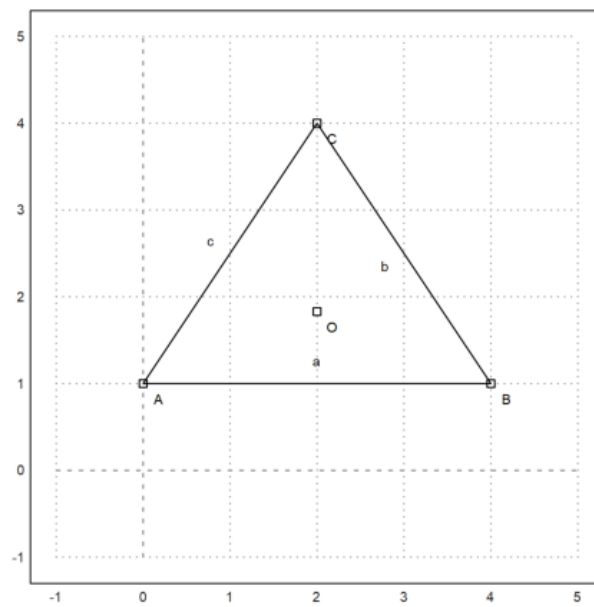
```
>plotSegment(B,C,"b"):
```



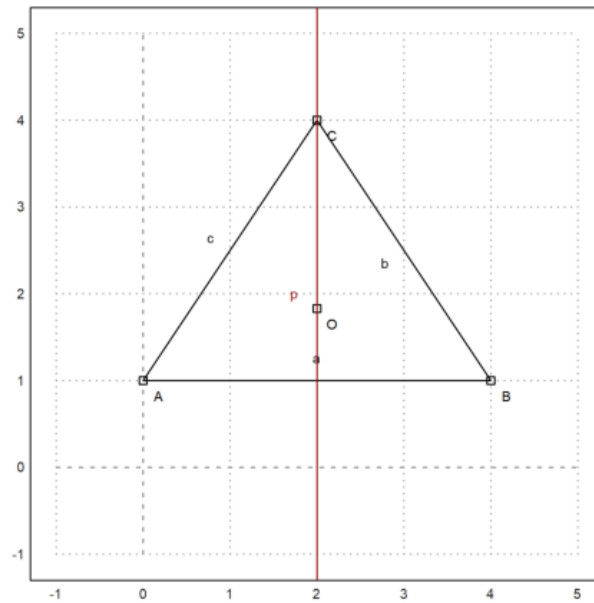
```
>plotSegment(A,C,"c"):
```



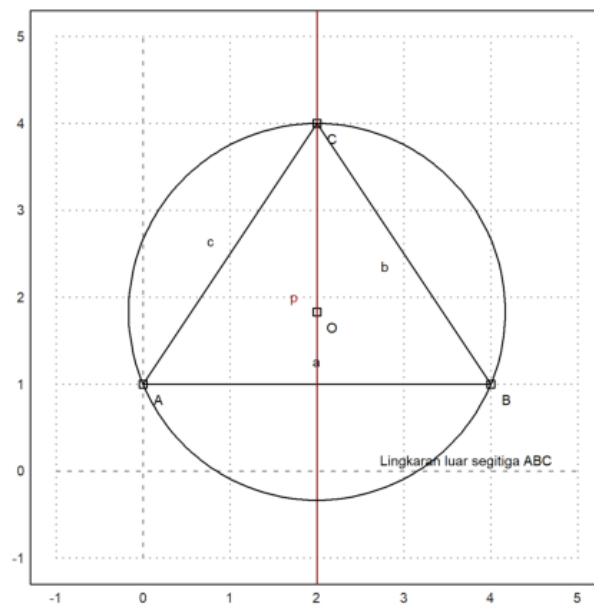
```
>c=circleThrough(A,B,C);
>R=getCircleRadius(c);
>O=getCircleCenter(c);
>plotPoint(O,"O"):
```



```
>l=angleBisector(A,C,B);
>color(2); plotLine(l); color(1):
```



```
>plotCircle(c,"Lingkaran luar segitiga ABC"):
```

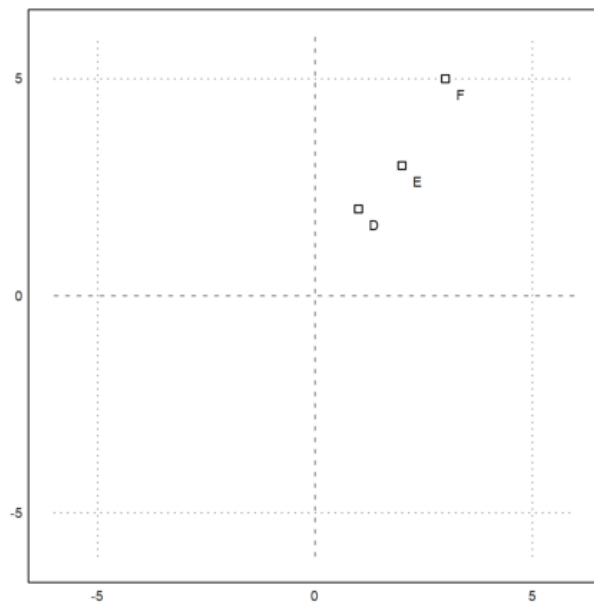


2. Gambarkan suatu parabola yang melalui 3 titik yang diketahui.

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;
>setPlotRange(6); D=[1,2]; E=[2,3]; F=[3,5];
>plotPoint(D,"D"); plotPoint(E,"E"); plotPoint(F,"F"):
```



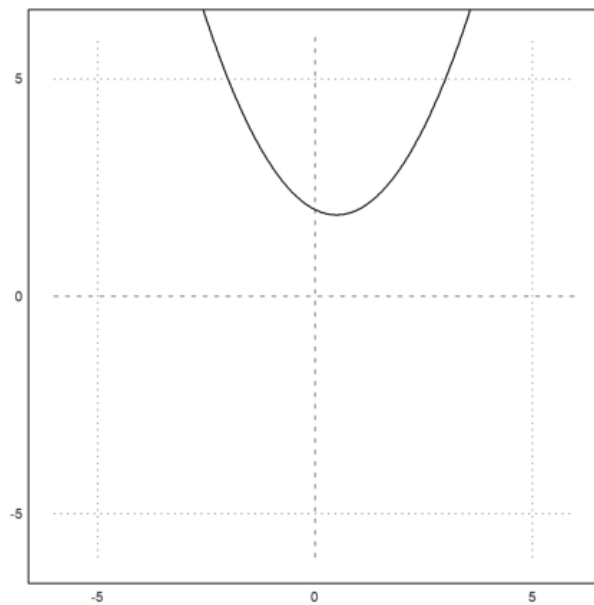
```
>sol &= solve([a+b+c=2,4*a+2*b+c=3,9*a+3*b+c=5],[a,b,c])
```

$$[[a = \frac{1}{2}, b = -\frac{1}{2}, c = 2]]$$

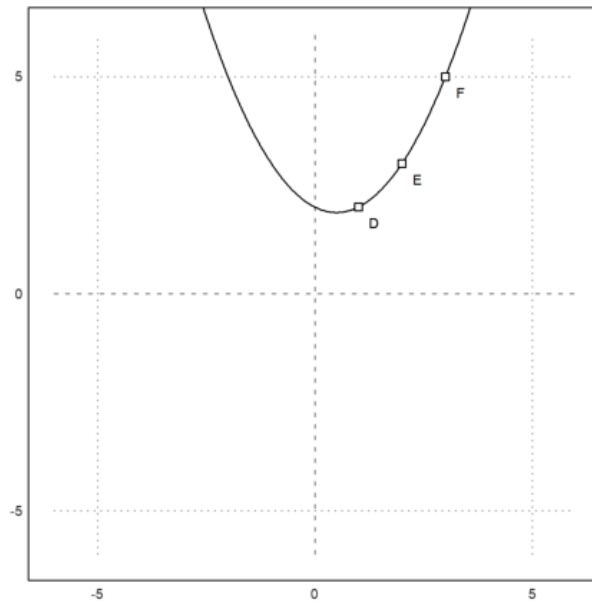
```
>function y&=(1/2)*x^2+(-1/2)*x+2
```

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

```
>plot2d("(1/2)*x^2+(-1/2)*x+2",-6,6,-6,6):
```



```
>plotPoint(D,"D"); plotPoint(E,"E"); plotPoint(F,"F"):
```

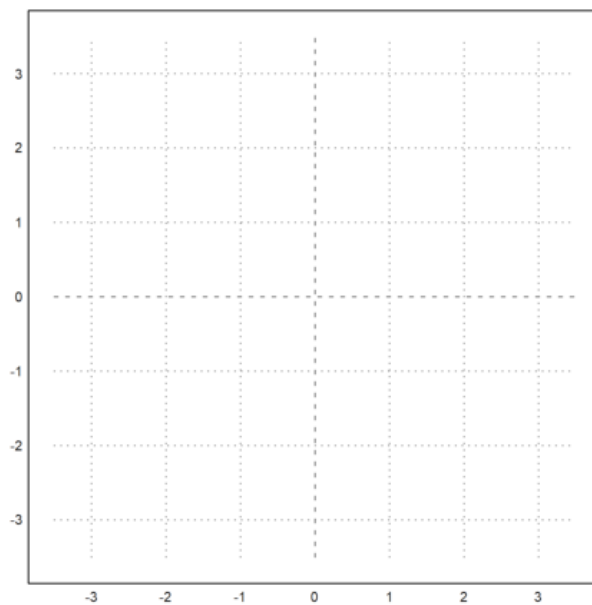


3. Gambarkanlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.
- Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung (sisinya-sisintya merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).
 - Suatu segi-4 merupakan segi-4 garis singgung apabila keempat garis bagi sudutnya bertemu di satu titik.
 - Jika segi-4 tersebut merupakan segi-4 garis singgung, gambar lingkaran dalamnya.
 - Tunjukkan bahwa syarat suatu segi-4 merupakan segi-4 garis singgung apabila hasil kali panjang sisi-sisi yang berhadapan sama.

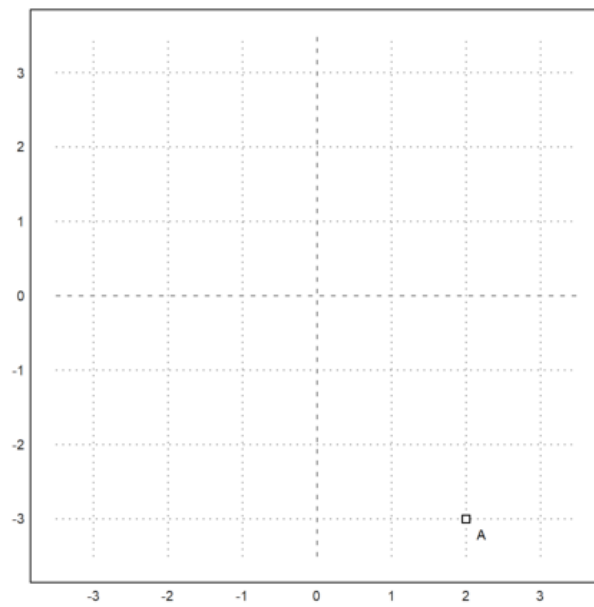
```
>load geometry
```

Numerical and symbolic geometry.

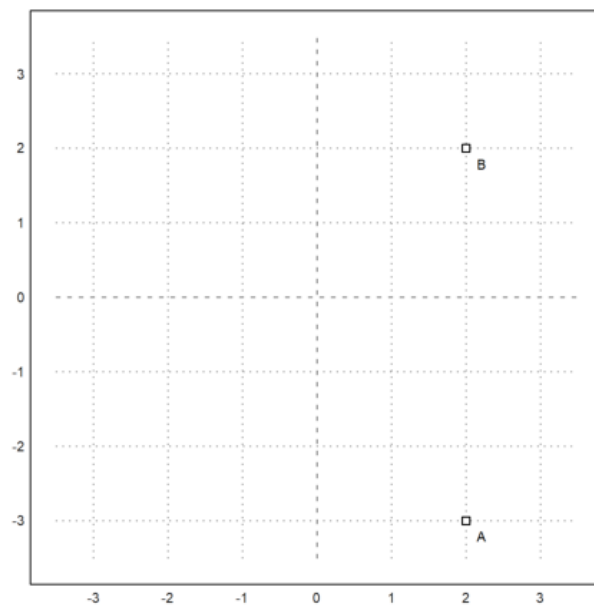
```
>setPlotRange(-3.5,3.5,-3.5,3.5):
```



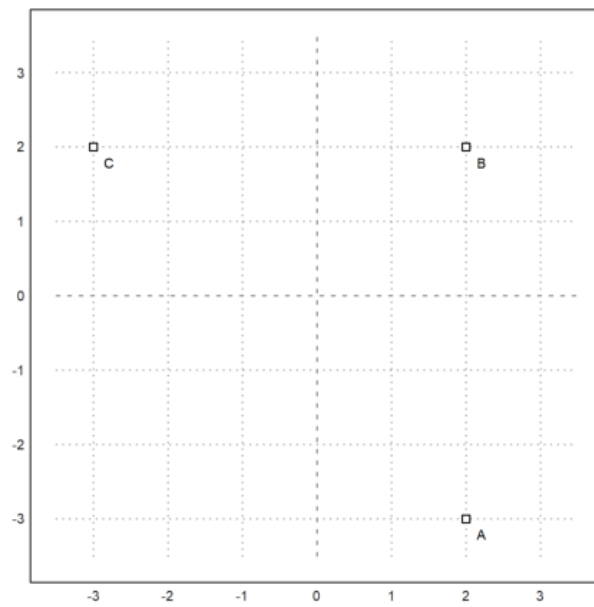
```
>A=[2,-3]; plotPoint(A,"A"):
```

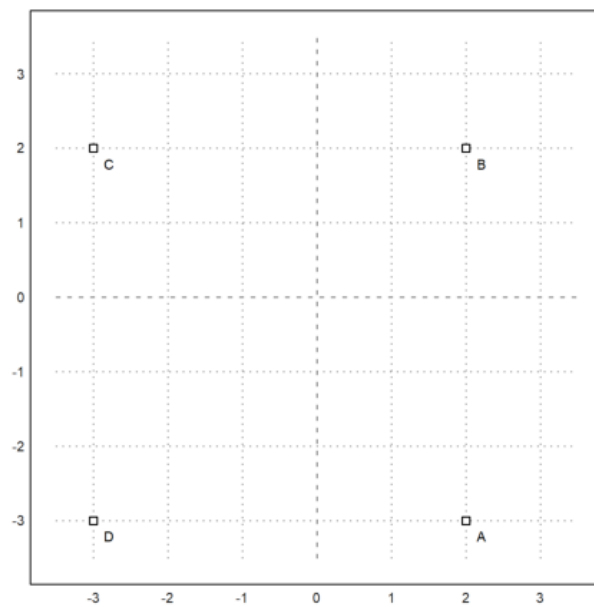
```
>B=[2,2]; plotPoint(B,"B"):
```



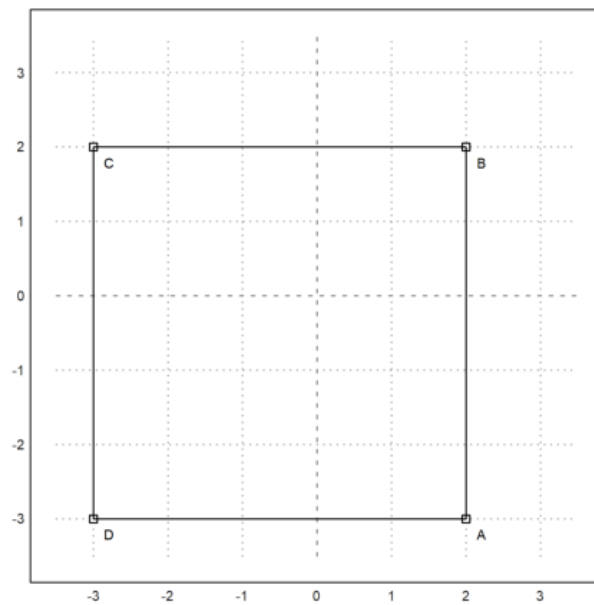
```
>C=[-3,2]; plotPoint(C,"C"):
```



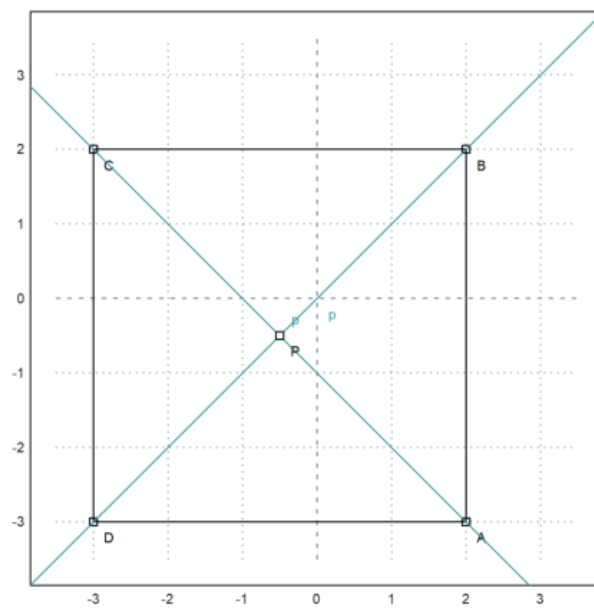
```
>D=[-3,-3]; plotPoint(D,"D"):
```



```
>plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,D,""); plotSegment(A,D,"");  
>aspect(1):
```

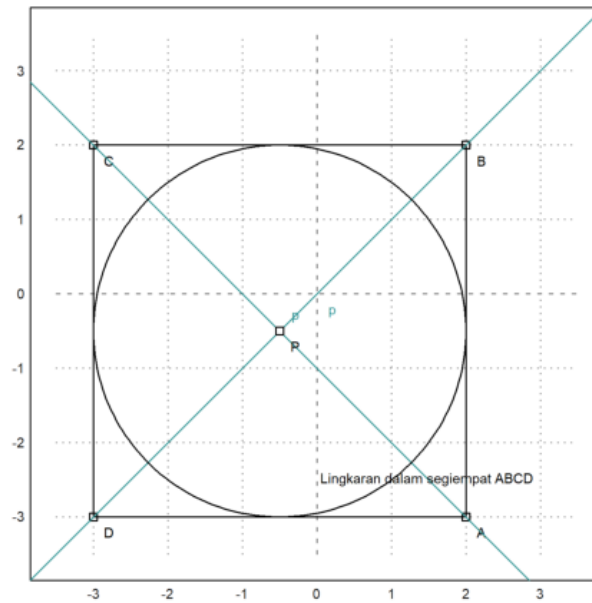


```
>l=angleBisector(A,B,C);
>m=angleBisector(B,C,D);
>P=lineIntersection(l,m);
>color(5); plotLine(l); plotLine(m); color(1);
>plotPoint(P,"P"):
```



Dapat disimpulkan bahwa keempat garis bagi sudutnya bertemu di satu titik yaitu di titik P.

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



Dibentuk lingkaran dalam segi empat yang ternyata sisi sisi dari segiempat merupakan garis-garis singgung lingkaran.

```
>AB=distance(A,B) //panjang sisi AB
```

5

```
>CD=distance(C,D) //panjang sisi CD
```

5

```
>AD=distance(A,D) //panjang sisi AD
```

5

```
>BC=distance(B,C) //panjang sisi BC
```

5

```
>AB.CD
```

25

```
>AD.BC
```

25

Terdapat,

$$AB \times CD = 25, AD \times BC = 25$$

Sehingga terbukti bahwa

$$AB \times CD, AD \times BC$$

Karena semua syarat telah terpenuhi, maka dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

```
>reset();
```

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).

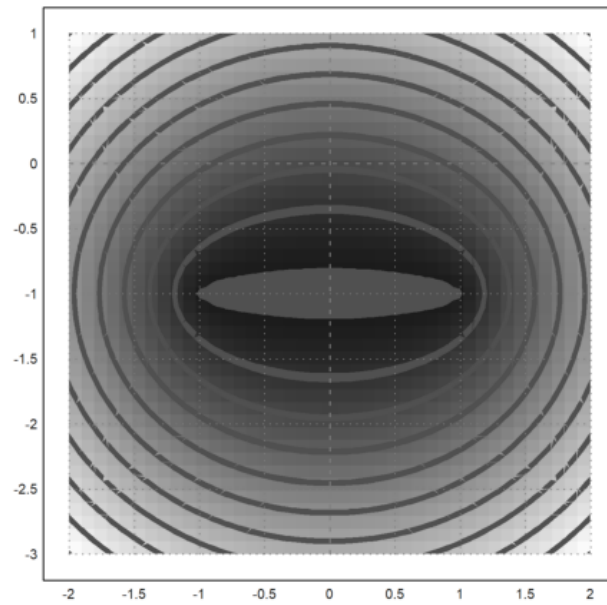
```
>load geometry
```

Numerical and symbolic geometry.

```

>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):

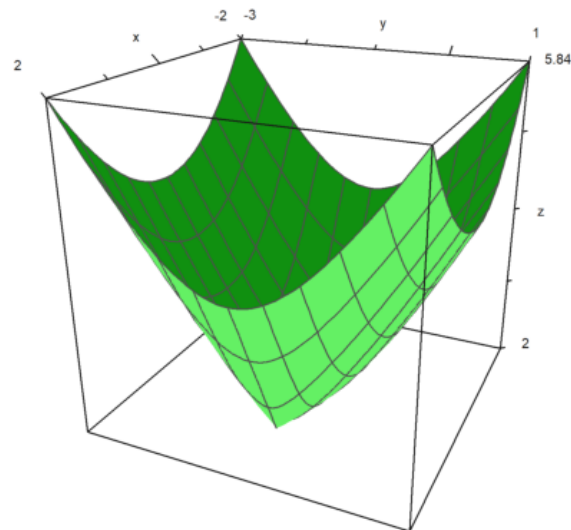
```



```

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

```

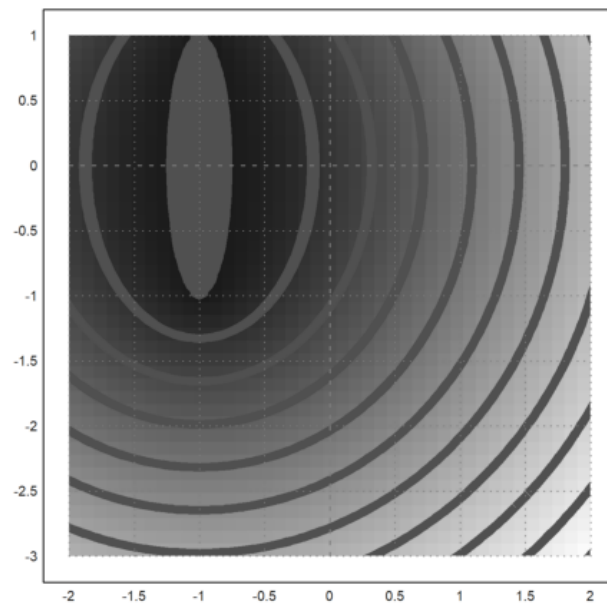


5. Gambarkanlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

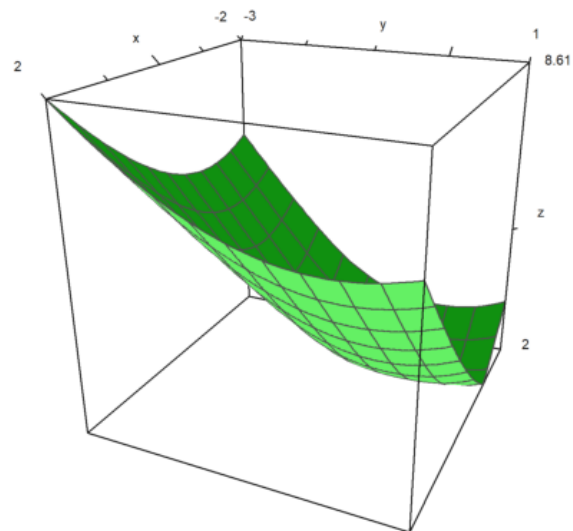
```

>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):
```

