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

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

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

Hitung garis tegak lurus melalui titik A pada BC.

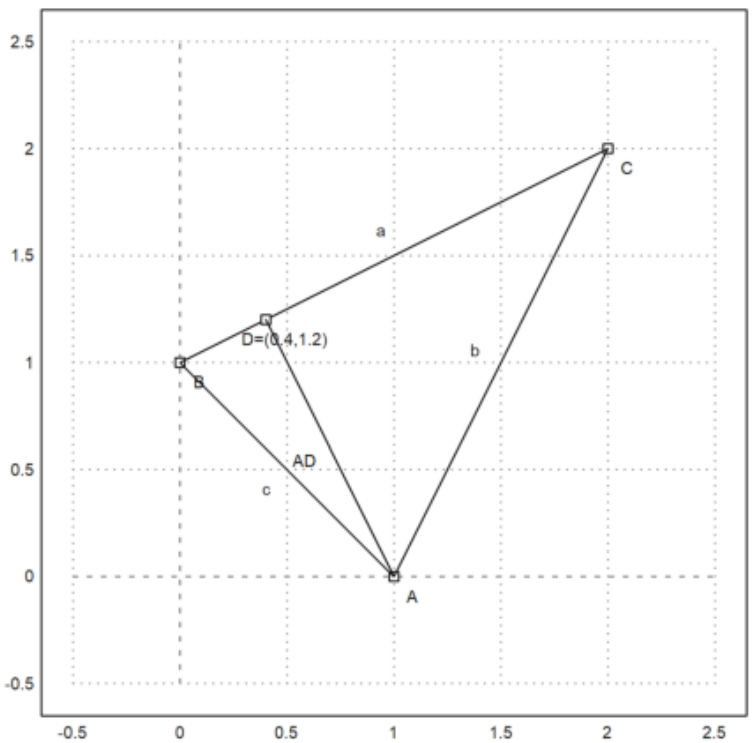
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan titik potongnya dengan BC.

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

Buat plotnya.

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



Hitung luas ABC:

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

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

```
  1.5
```

BAndingkan menggunakan rumus determinan

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

```
  1.5
```

Cara lain menghitung luas segitigas ABC:

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

```
  1.5
```

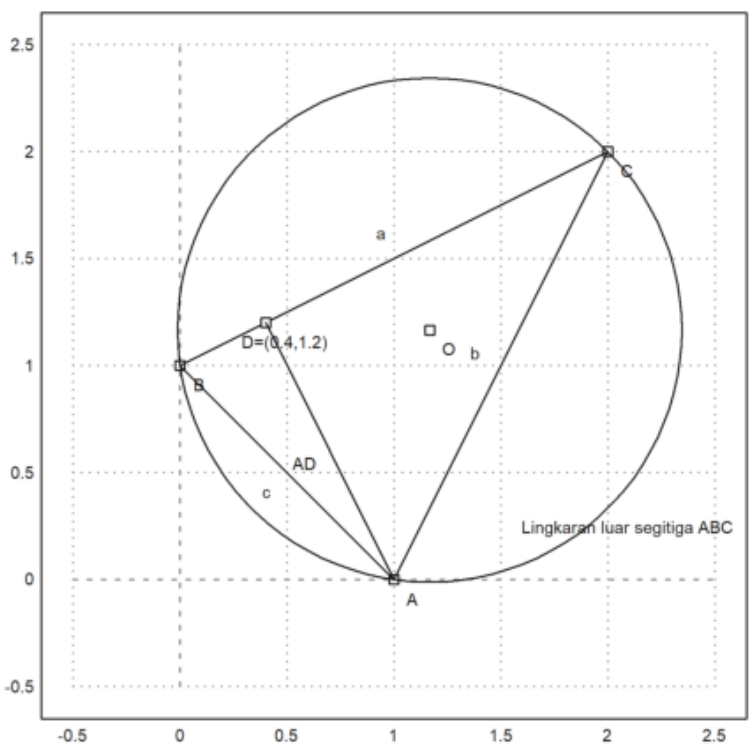
Sudut pada titik C.

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

```
  36°52’11.63’’
```

Sekarang lingkaran keliling (circumcircle) 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,]
  1.17851130198
```

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
```

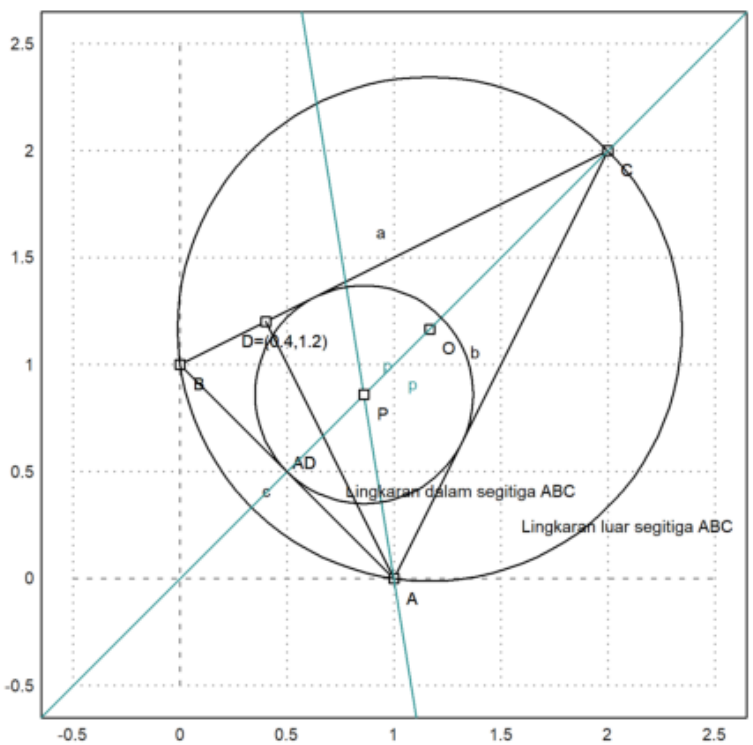
```
  [0.86038,  0.86038]
```

Tambahkan semuanya ke plotnya.

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

```
  0.509653732104
```

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



Latihan

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

```
>$A:=[1,0]
```

$$[1,0]$$

```
>$B:=[0,1]
```

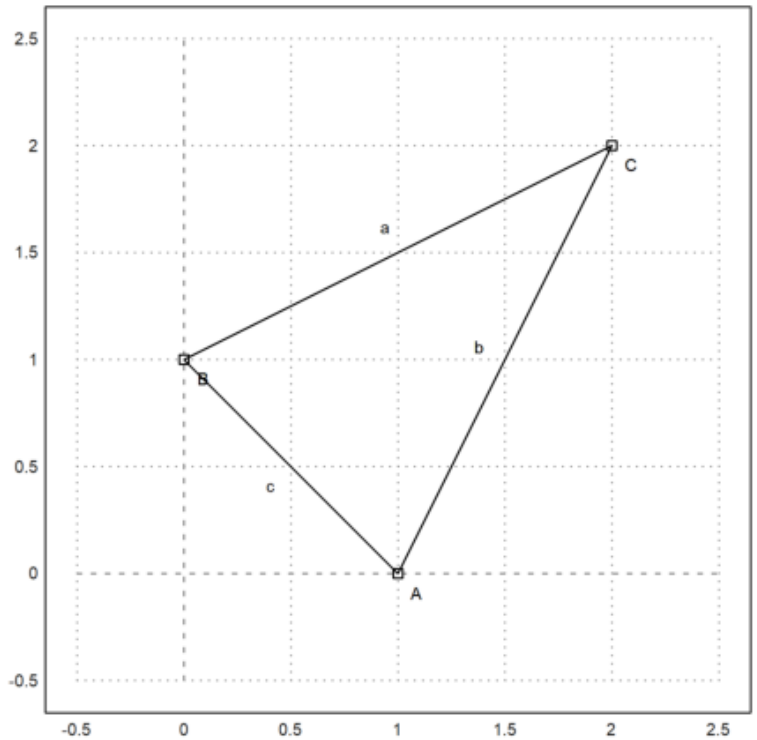
$$[0,1]$$

```
>$C:=[2,2]
```

$$[2,2]$$

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

```
>setPlotRange(-0.5,2.5,-0.5,2.5);
>A=[1,0]; plotPoint(A,"A");
>B=[0,1]; plotPoint(B,"B");
>C=[2,2]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b"):
```



Segitiga yang terbentuk merupakan segitiga sama kaki.

3. Hitung luas segitiga tersebut.

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

```
  1.5
```

```
>
```

Contoh 2: Geometri Simbolik

Kita dapat menghitung geometri secara tepat dan simbolik menggunakan Maxima.

File geometry.e menyediakan fungsi-fungsi yang sama (dan bahkan lebih banyak) di Maxima. Namun, kini 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 pada Euler, tetapi mendukung perhitungan simbolik.

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

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

Kita dapat memperoleh persamaan sebuah garis dengan mudah.

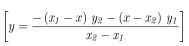
```
>$getLineEquation(c,x,y), $solve(%,y) | expand // persamaan garis c
```

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



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

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



```
>$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
```

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

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

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

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

$$\sqrt{\left(\frac{6}{5}-jA\right)^2+\left(\frac{2}{5}-jA\right)^2}$$

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

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

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

$$1.178511301977579$$

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

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

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

$$y=x$$

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

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

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

```
  [0.86038,  0.86038]
```

Garis dan Lingkaran yang Berpotongan

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

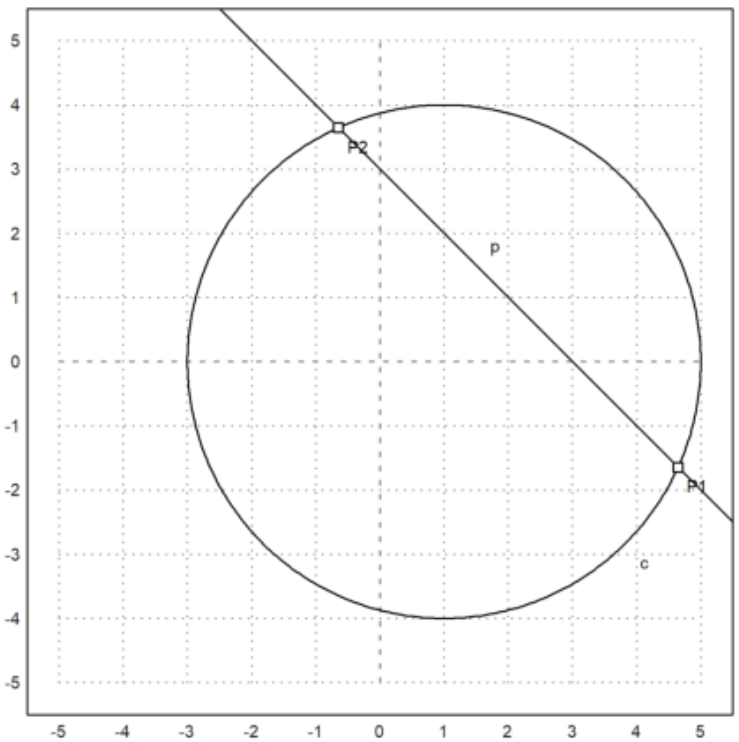
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

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

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

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

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



Hal yang sama berlaku di Maxima.

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

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

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

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

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

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

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

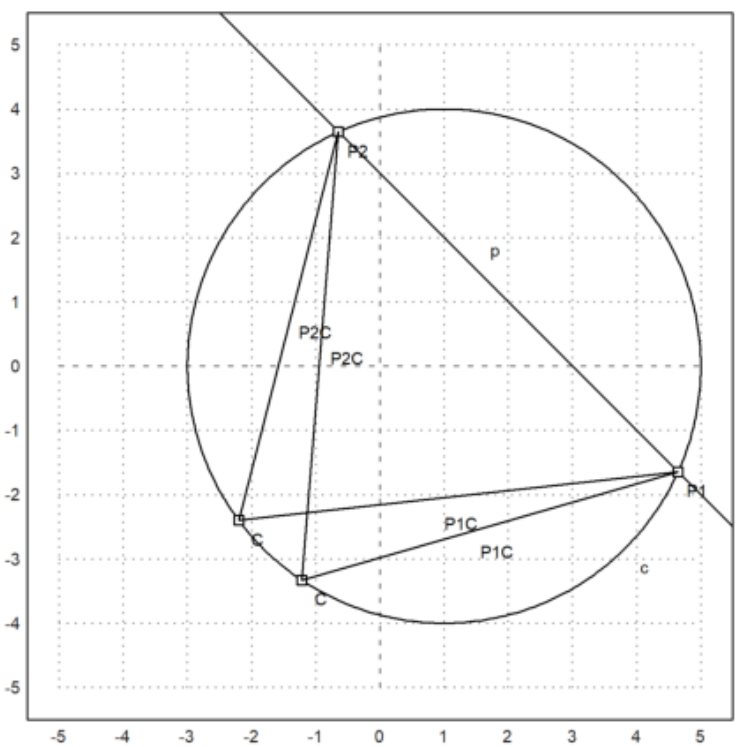
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
>degprint(computeAngle(P1,C,P2))
```

```
  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;
```

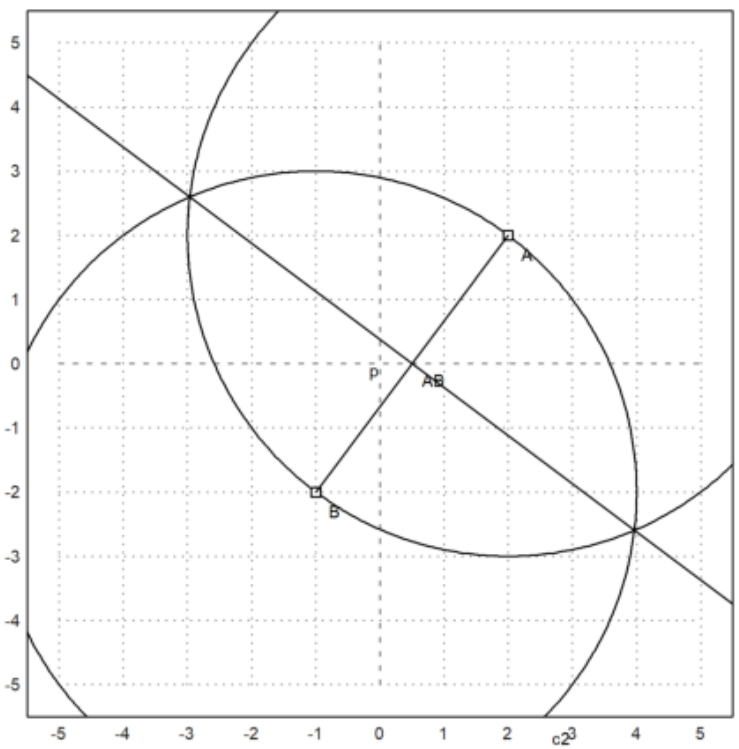


Garis Sumbu

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 melakukan hal yang sama di Maxima dengan koordinat umum.

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

Persamaan untuk titik potongnya cukup rumit. Namun, kita dapat menyederhanakannya jika kita menyelesaikannya terhadap 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 garis sumbu 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:

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

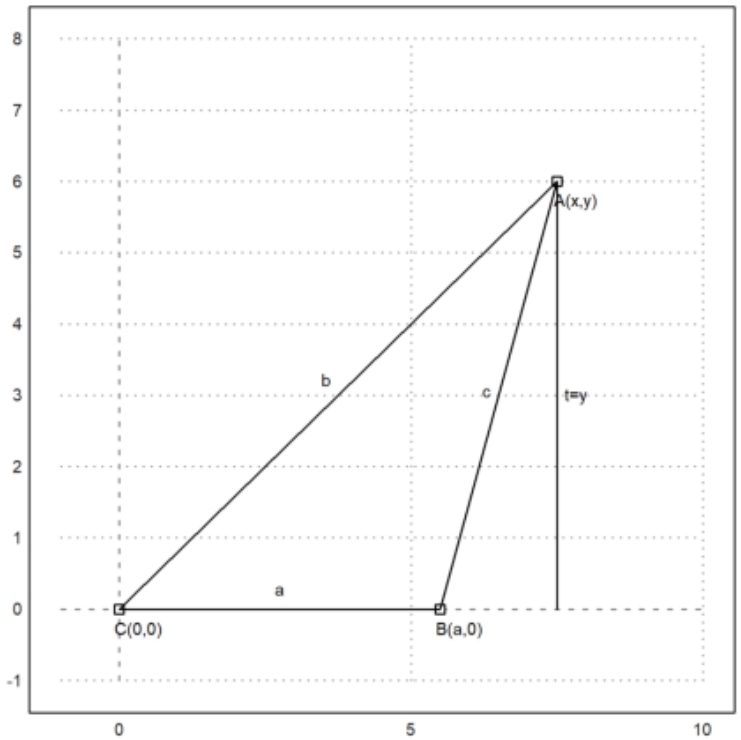
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

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])
```

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

Extract the solution y.

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

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

Ekstrak solusi y.

```
>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
```

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

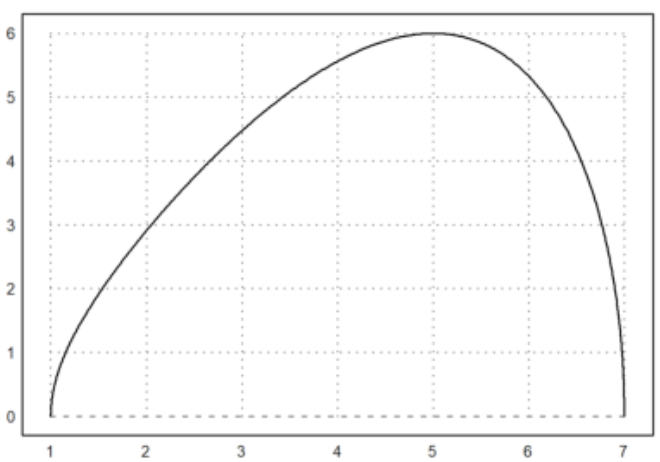
Tentu saja, setiap segitiga siku-siku merupakan kasus yang sudah dikenal dengan baik.

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

```
  6
```

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

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



Kasus umum juga dapat diterapkan.

```
>$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. Telah diketahui bahwa ini merupakan sebuah elips.

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

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

Lalu buat fungsi dari ini.

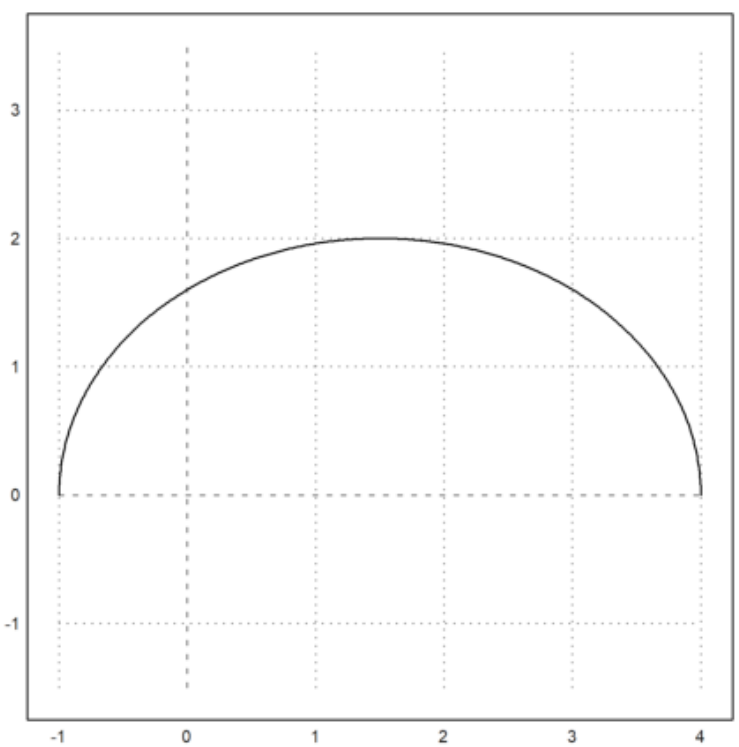
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d
```

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

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

Sekarang kita dapat menggambar himpunannya. Sisi b bervariasi dari 1 hingga 4. Telah diketahui bahwa hasilnya adalah 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

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

di mana (xm,ym) adalah pusatnya, dan u serta v merupakan setengah sumbu elips.

```
>$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, mencapai nilai maksimum saat x=0. Jadi, luas segitiga dengan a+b+c=d adalah maksimum jika segitiga tersebut sama sisi. Kita ingin menurunkannya secara analitik.

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

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

Kita mendapatkan beberapa nilai minimum, yang merupakan segitiga dengan satu sisi bernilai 0, serta solusi a=b=c=d/3.

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

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

Terdapat juga metode Lagrange, yaitu memaksimalkan H(a,b,c)^2 dengan syarat 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])
```

```
                      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 situasinya

Pertama, tetapkan titik-titiknya di Maxima.

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

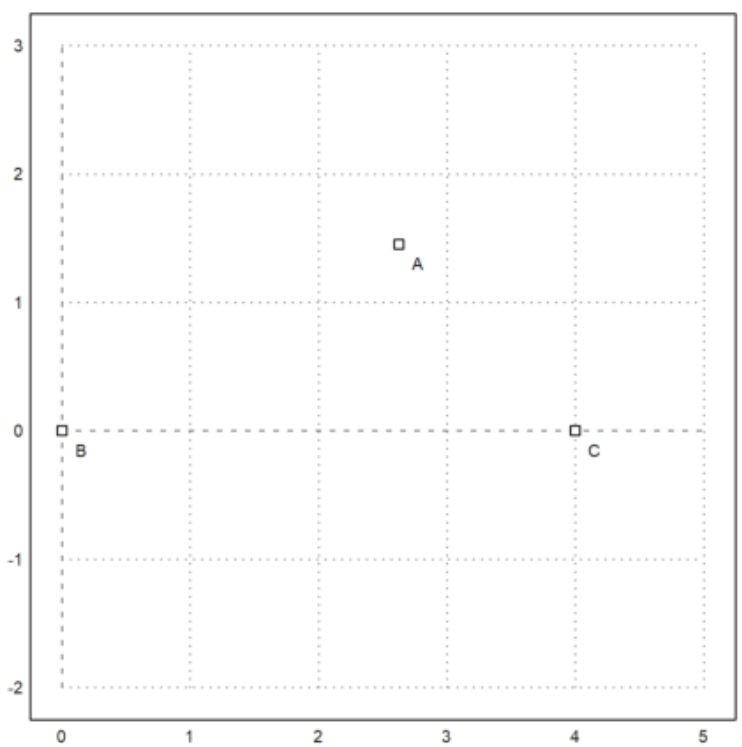
$$\left[\frac{-c^2+b^2+a^2}{2\,a},\frac{\sqrt{-c^4+2\,b^2\,c^2+2\,a^2\,c^2-b^4+2\,a^2\,b^2-a^4}}{2\,a}\right]$$

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

$$[a,0]$$

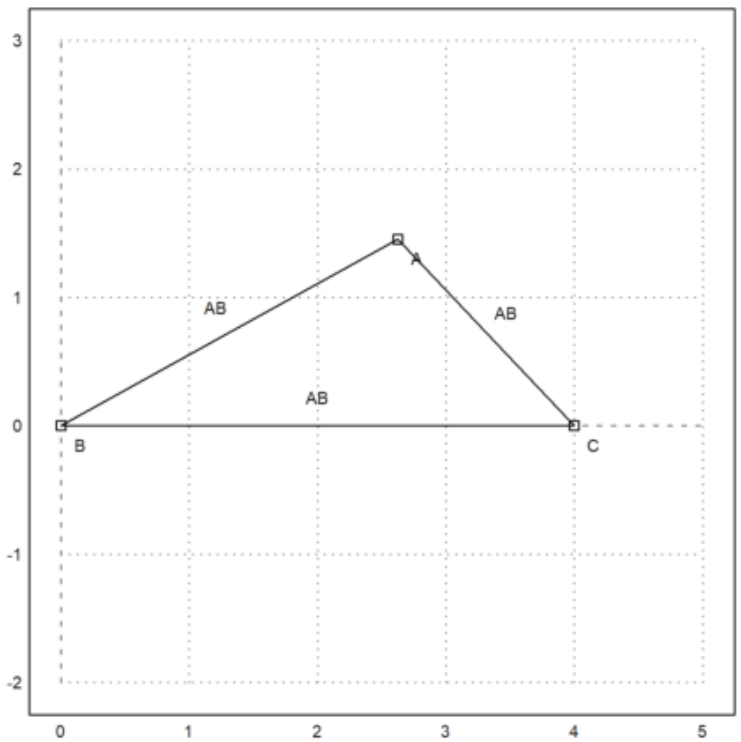
Kemudian atur jangkauan plot, dan gambarkan titik-titiknya.

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



Plot segmen-segmennya.

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



Hitung garis sumbu tegak lurus tengah di Maxima.

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

Dan pusat lingkaran keliling.

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

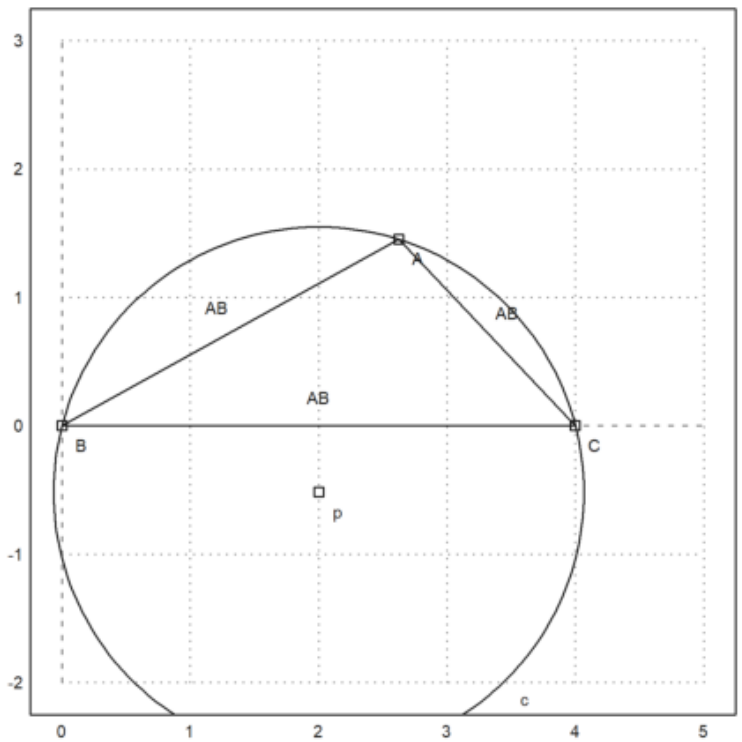
Kita mendapatkan rumus untuk jari-jari lingkaran keliling.

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

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

Mari kita tambahkan ini ke dalam plot.

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



Dengan menggunakan geometri, kita dapat menurunkan rumus sederhana

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

untuk jari-jari. Kita dapat memeriksa apakah hal ini benar dengan Maxima. Maxima hanya akan memfaktorkannya jika kita mengkuadratkan persamaan tersebut.

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

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

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, lingkaran keliling, titik berat, titik Exeter, dan pusat lingkaran sembilan titik dari segitiga.

Sebagai demonstrasi, kita akan menghitung dan menggambar garis Euler pada sebuah segitiga.

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

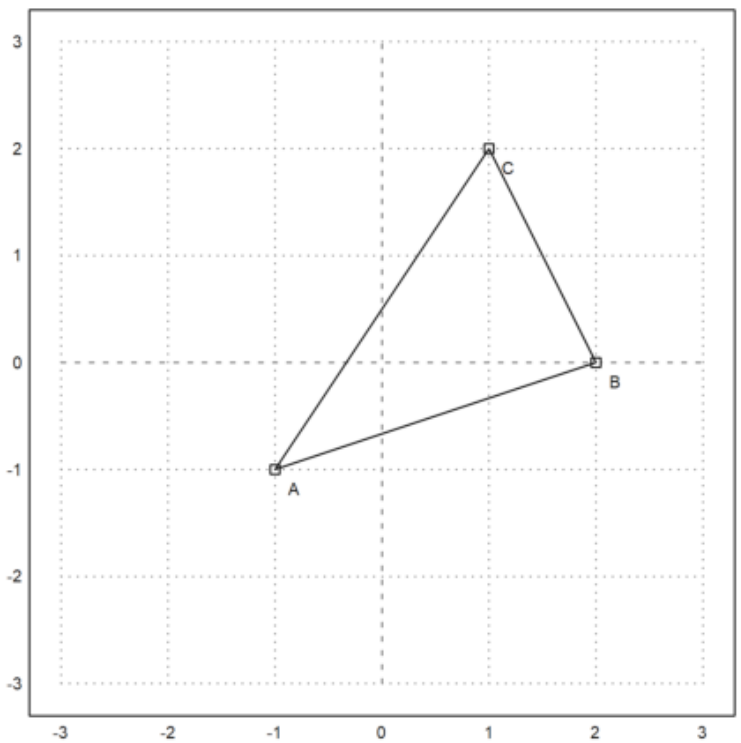
```
>A::=[-1,-1]; B::=[2,0]; C::=[1,2];
```

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

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

Kita juga dapat menambahkan sisi-sisi segitiga tersebut.

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



Berikut adalah luas segitiga tersebut dengan 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. Menyisipkan 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 keliling dari ABC.

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

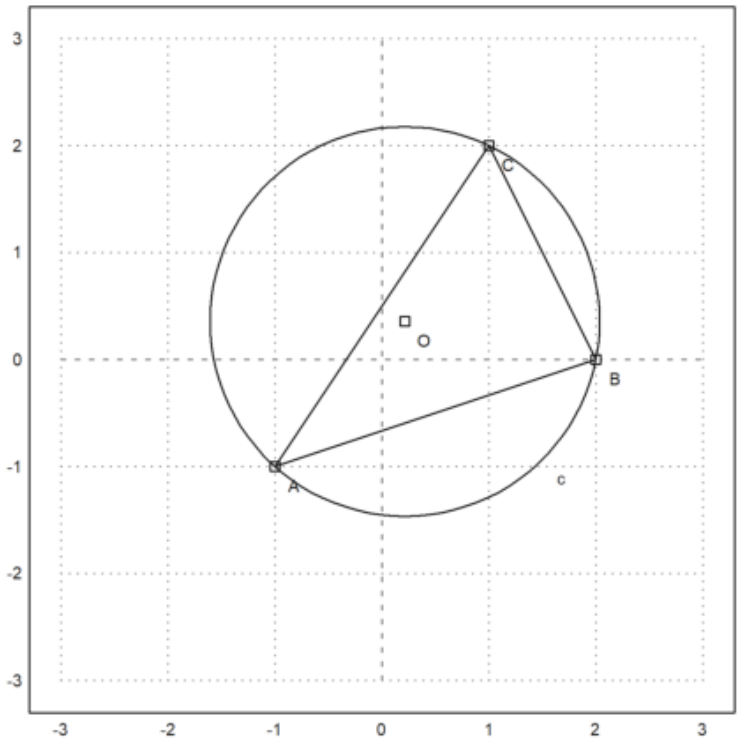
$$\left(y-\frac{5}{14}\right)^2+\left(x-\frac{3}{14}\right)^2=\frac{325}{98}$$

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

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

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

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



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

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

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

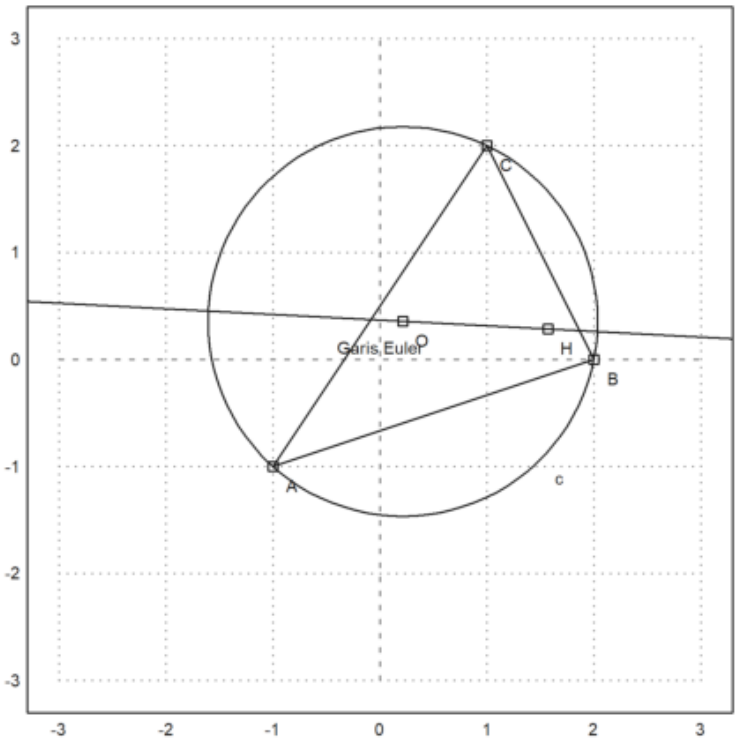
Sekarang kita dapat menghitung garis Euler dari segitiga tersebut.

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

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

Tambahkan ke plot kita.

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

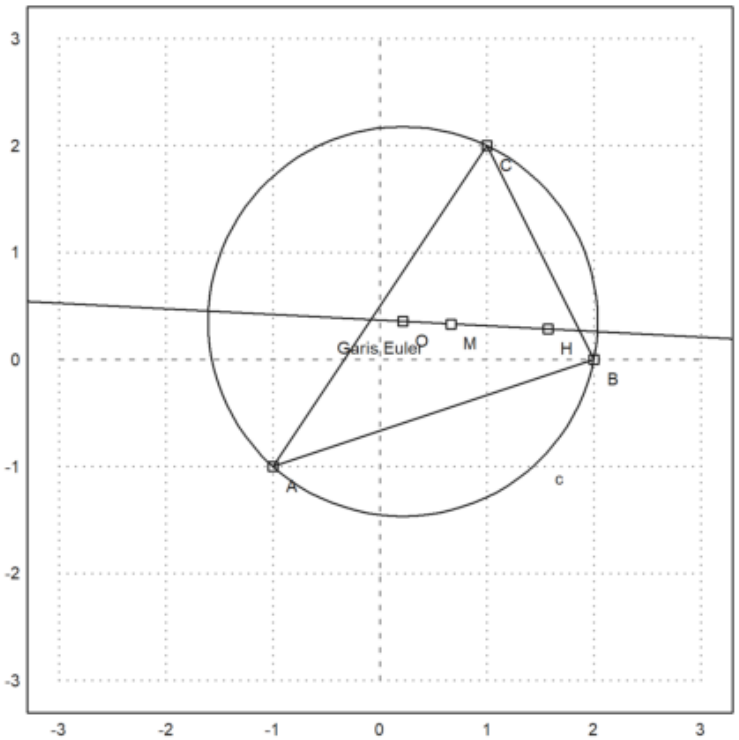


Titik pusat gravitasi seharusnya berada di garis ini.

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

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

$$2$$

Fungsi-fungsi ini juga mencakup fungsi untuk sudut.

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

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

```
  60°15’18.43’’
```

Persamaan untuk pusat lingkaran dalam tidak terlalu rapi.

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

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

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

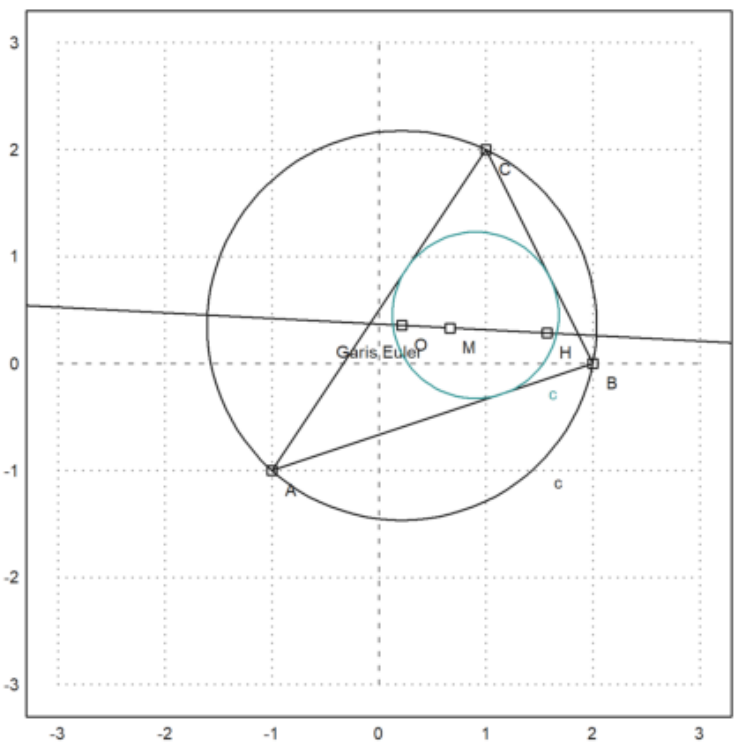
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
```

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

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

Mari kita tambahkan ini ke dalam plot.

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



Parabola

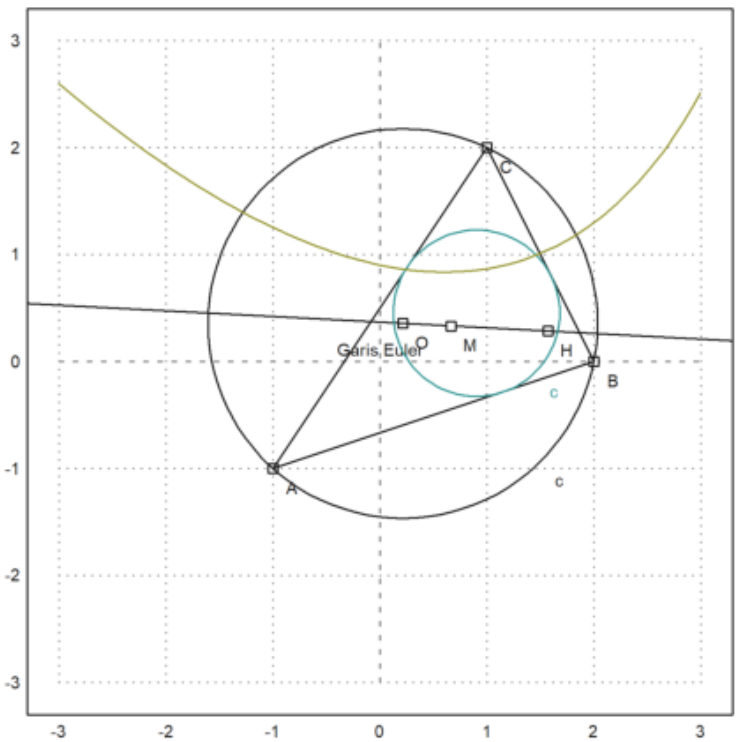
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Ini seharusnya merupakan sebuah fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya 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)
```

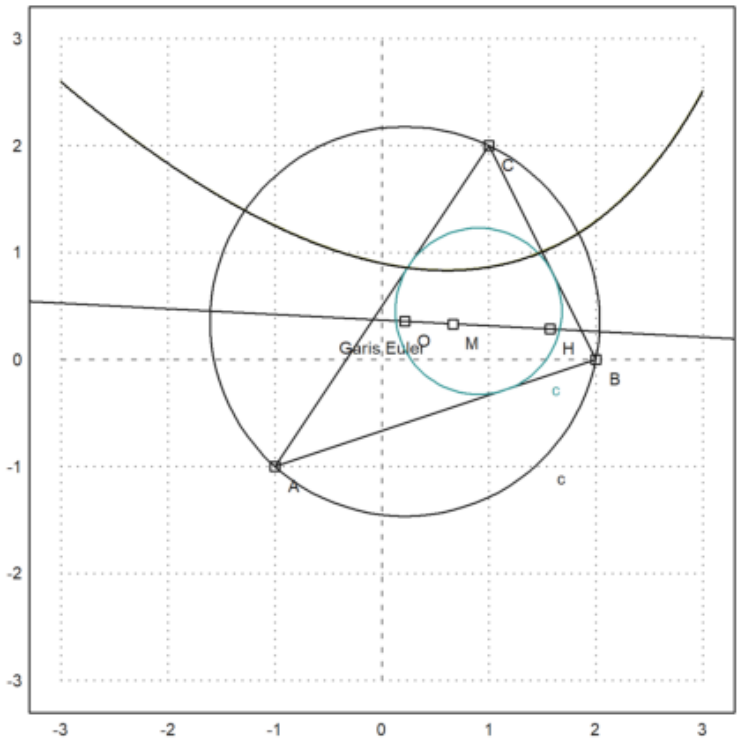
$$\left[y = -3\,x - \sqrt{70}\,\sqrt{9 - 2\,x} + 26,\ y = -3\,x + \sqrt{70}\,\sqrt{9 - 2\,x} + 26\right]$$

Solusi pertama adalah

$$akar_1$$

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

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

```
>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 dari sebuah ceramah N.J. Wildberger. Dalam bukunya "Divine Proportions", Wildberger mengusulkan untuk menggantikan konsep klasik jarak dan sudut dengan kuadran dan spread. Dengan menggunakan ini, memang memungkinkan untuk menghindari fungsi trigonometri pada banyak contoh, dan tetap "rasional".

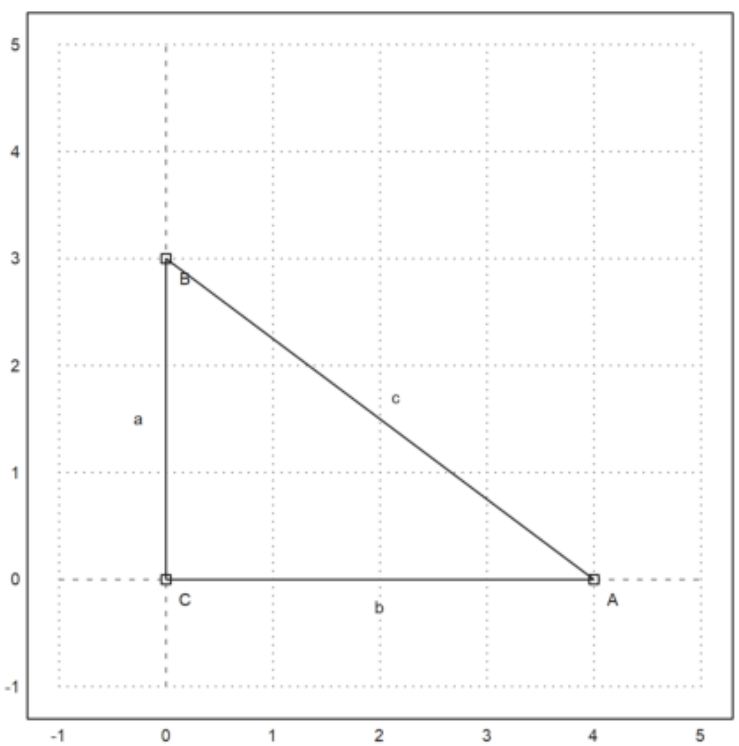
Berikut ini, saya memperkenalkan konsep-konsep tersebut, dan menyelesaikan beberapa masalah. Saya menggunakan perhitungan simbolik Maxima di sini, yang menyembunyikan keuntungan utama dari trigonometri rasional yaitu perhitungan dapat dilakukan hanya dengan kertas dan pensil. Anda diundang untuk memeriksa 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 aproksimasi numerik.

```
>load geometry;
```

Untuk pengenalan pertama, kita menggunakan segitiga siku-siku dengan perbandingan Mesir yang terkenal 3, 4, dan 5. Perintah-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 dari fungsi sinus. Hasilnya adalah sudut yang sulit dicerna, yang hanya dapat dicetak secara aproksimasi.

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

```
  36°52’11.63’’
```

Trigonometri rasional berusaha menghindari hal ini.

Konsep pertama dari trigonometri rasional adalah kuadran, yang menggantikan jarak. Sebenarnya, ini hanyalah jarak yang dikuadratkan. Berikut ini, a, b, dan c menyatakan kuadran dari sisi-sisinya.

Teorema Pythagoras kemudian cukup menjadi a+b=c.

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

$$25 = 25$$

Konsep kedua dari trigonometri rasional adalah spread. Spread mengukur bukaan antara garis-garis. Nilainya 0 jika garis-garis sejajar, dan 1 jika garis-garis tegak lurus. Ini merupakan kuadrat dari sinus sudut antara kedua garis.

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 kuadran dari setiap segitiga siku-siku dengan satu sudut di A.

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

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

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

Tentu saja, kita dapat mengubah nilai aproksimasi sudut wa menjadi spread, dan mencetaknya sebagai pecahan.

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

```
  9/25
```

Hukum kosinus dalam 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 adalah spread di sudut A. Sisi a, seperti biasa, berada di seberang sudut A.

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

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

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

Dalam kasus kita, kita mendapatkan

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

$$1024=1024$$

Mari kita gunakan hukum silang ini untuk menemukan spread di A. Untuk melakukan ini, kita buat hukum silang untuk kuadran a, b, dan c, dan menyelesaikannya untuk spread yang tidak diketahui sa.

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

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

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

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

Kita sudah mengetahuinya. Definisi spread adalah kasus khusus dari hukum silang.

Kita juga dapat menyelesaikannya untuk a, b, c secara umum. Hasilnya adalah sebuah rumus yang menghitung spread suatu sudut segitiga jika diketahui kuadran dari ketiga sisinya.

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

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

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

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

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

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

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

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

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

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

Ini adalah sudut dalam derajat.

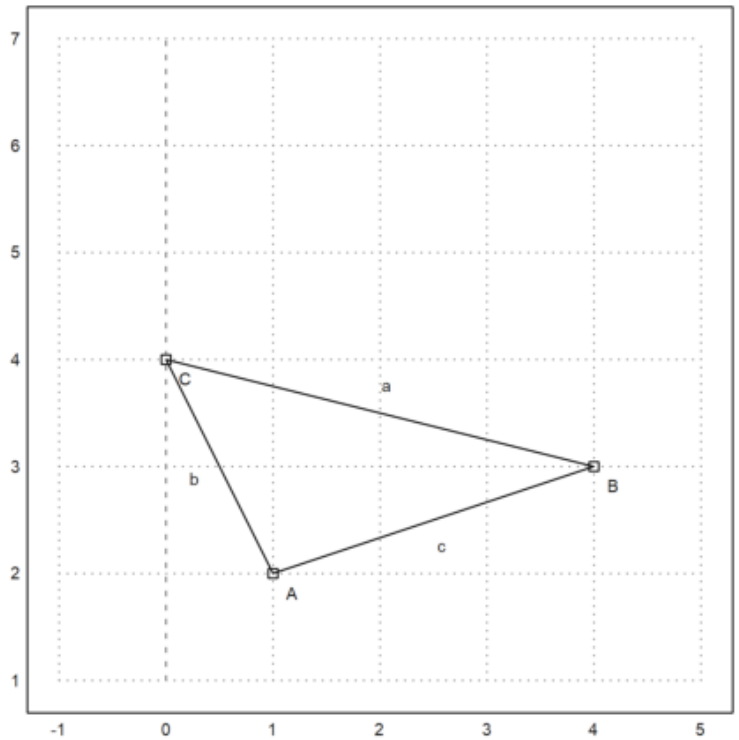
```
>degprint(arcsin(sqrt(6/7)))
```

```
  67°47’32.44’’
```

Contoh 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 Pythagoras, mudah untuk menghitung jarak antara dua titik. Saya pertama-tama menggunakan fungsi distance dari file Euler untuk geometri. Fungsi distance menggunakan geometri klasik.

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

$$\sqrt{10}$$

Euler juga berisi fungsi untuk kuadran antara dua titik.
Dalam contoh berikut, karena c+b tidak sama dengan a, segitiga tersebut bukan segitiga siku-siku.

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


Pertama, mari kita hitung sudut tradisional. Fungsi computeAngle menggunakan metode biasa berdasarkan perkalian titik dari dua vektor. Hasilnya adalah aproksimasi 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 dapat melakukan hal yang sama dengan hukum silang. Kita masukkan kuadran 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)
```

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



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. Ia menyelesaikan istilah sin(arccos(...)) menjadi hasil pecahan. Sebagian besar siswa tidak bisa melakukan ini.

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

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

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

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

menurut definisi.

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

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

Gambar berikut dibuat dengan program geometri C.a.R., yang dapat menggambar kuadran dan spread.
image: (20) Rational_Geometry_CaR.png
Menurut definisi, panjang ha adalah akar kuadrat dari kuadrannya.

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

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

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, kita bekerja dengan kuadran!

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

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

Rumus determinan biasa memberikan hasil yang sama.

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

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

Rumus Heron

Sekarang, mari kita selesaikan masalah ini secara umum!

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

Pertama, kita menghitung spread di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat (atau "quadrea"?), memfaktorkannya dengan Maxima, dan kita mendapatkan rumus terkenal Heron. Memang, ini sulit dilakukan dengan kertas dan pensil.

```
>$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 mereka tidak dapat dijumlahkan begitu saja seperti sudut.
Namun, tiga spread dari sebuah segitiga memenuhi aturan "triple spread" berikut.

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

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

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

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

Karena spread dari

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

sama, aturan triple spread juga berlaku jika

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

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

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

Sebagai contoh, kita dapat menghitung spread dari sudut 60°. Nilainya 3/4. Namun, persamaan ini memiliki solusi kedua, di mana semua spread bernilai 0.

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

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

Spread dari sudut 90° jelas 1. Jika dua sudut dijumlahkan menjadi 90°, spread mereka menyelesaikan persamaan triple spread dengan a,b,1. Dari perhitungan berikut, kita mendapatkan a+b=1.

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

$$[x=1-y]$$

Karena spread dari 180°-t sama dengan spread dari t, rumus triple spread juga berlaku jika satu sudut merupakan jumlah atau selisih dari dua sudut lainnya.
Jadi kita dapat menemukan spread dari sudut yang digandakan. Perlu dicatat bahwa ada dua solusi lagi. Kita jadikan ini sebagai sebuah fungsi.

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

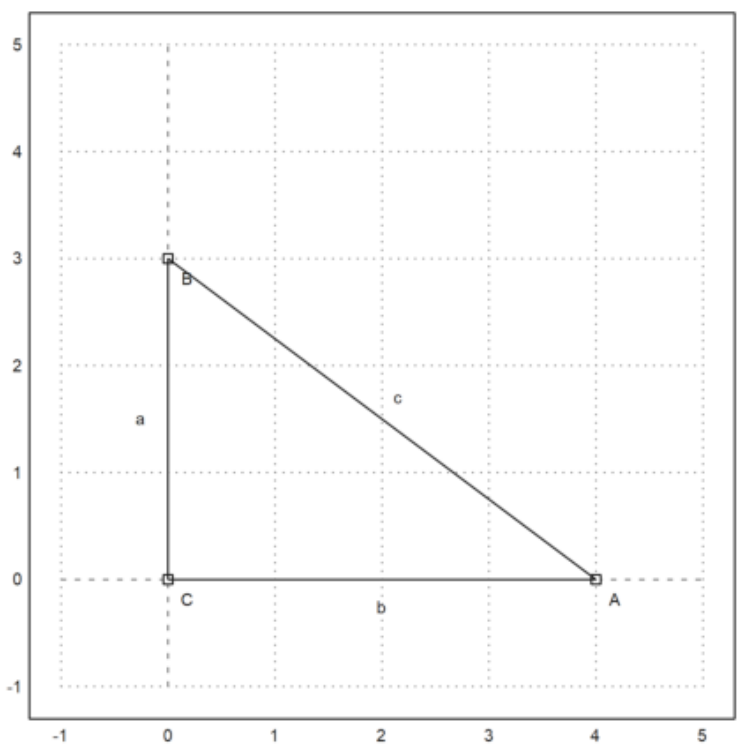
$$[x = 4\,a - 4\,a^2, x = 0]$$

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

Pembagi Sudut

Ini adalah situasi 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 garis bagi sudut di A. Namun, kita ingin menyelesaikannya untuk a, b, c secara umum.

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

Jadi kita pertama-tama menghitung spread dari sudut yang dibagi di 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
```

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

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

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

Mari kita periksa untuk persegi panjang Mesir.

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

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

Kita dapat mencetak sudut di Euler, setelah mengubah spread menjadi radian.

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

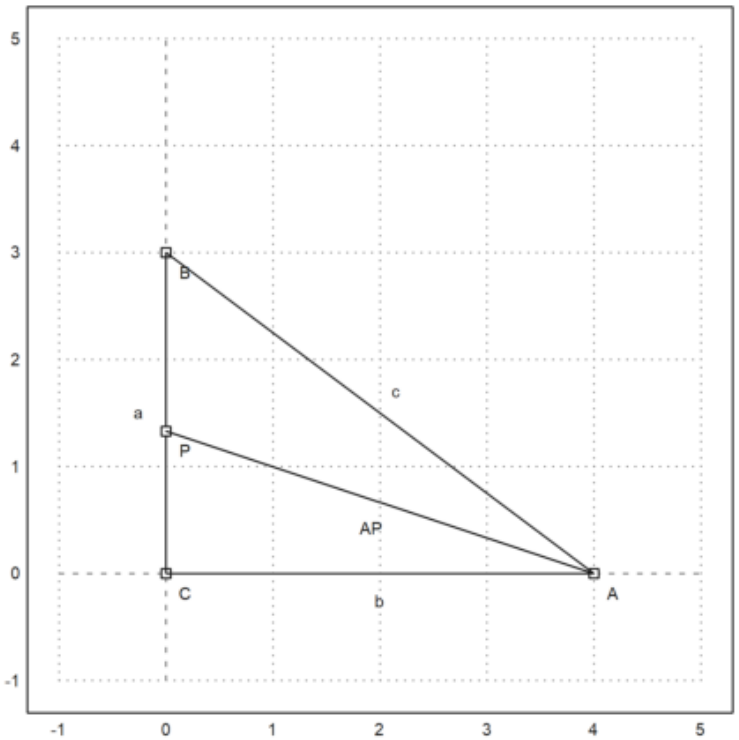
```
  18°26’5.82’’
```

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.

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

```
  [0,  1.33333]
```

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



Mari kita periksa sudut-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 dalam segitiga APC. Teorema ini menyatakan bahwa

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

berlaku dalam segitiga manapun. Jika dikuadratkan, ini diterjemahkan menjadi hukum yang disebut "spread law"

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

di mana a,b,c menyatakan kuadran.

Karena spread CPA adalah 1-sa2, kita mendapatkan dari sini bisa/1=b/(1-sa2) dan dapat menghitung bisa (kuadran dari garis bagi sudut).

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

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

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
```

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

Nilainya sama dengan yang kita dapatkan menggunakan rumus trigonometri.

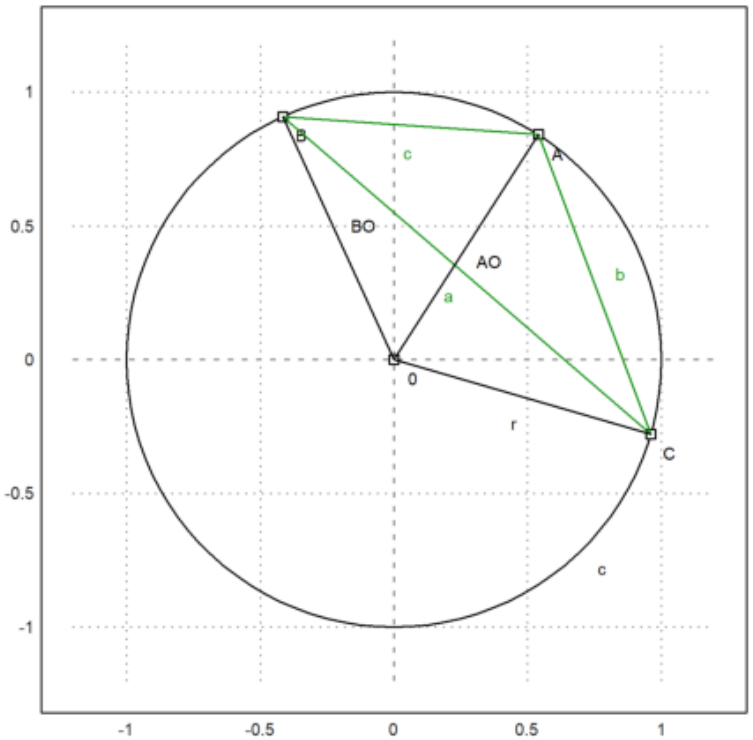
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

```
  1.33333333333
```

Sudut Tali Busur

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,"0"); ...
>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 lingkaran luar dalam hal kuadran sisi-sisinya.

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
```

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

Kita dapat menjadikan ini sebagai fungsi di Euler.

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

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

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

Jari-jarinya memang 1.

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

```
  1
```

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

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

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

Sebenarnya, spread 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
```

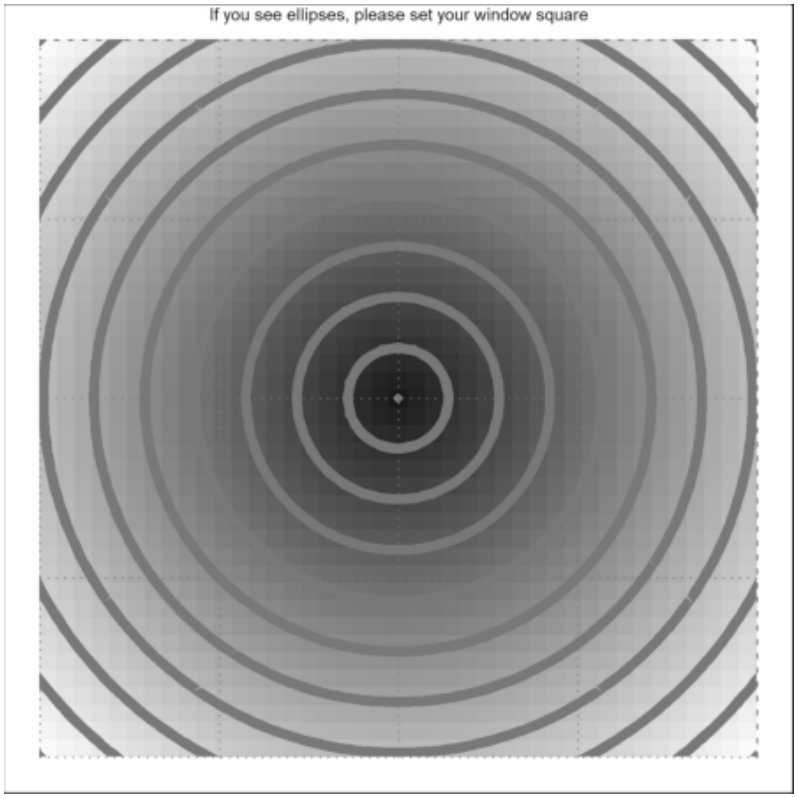
$$0$$

Contoh 6: Jarak Minimal pada Bidang

Catatan Pendahuluan

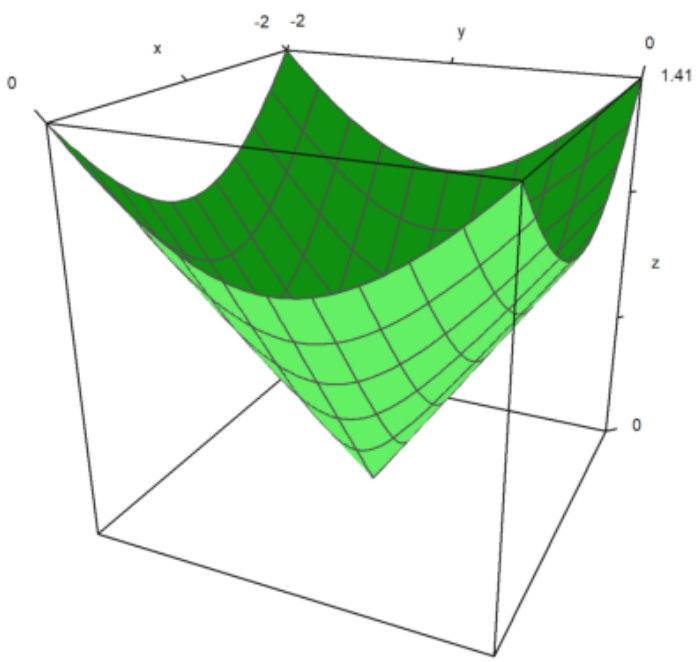
Fungsi yang, untuk sebuah titik M di bidang, menentukan jarak AM antara titik tetap A dan M, memiliki garis level yang cukup sederhana: lingkaran dengan pusat di A.

```
>&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):
```

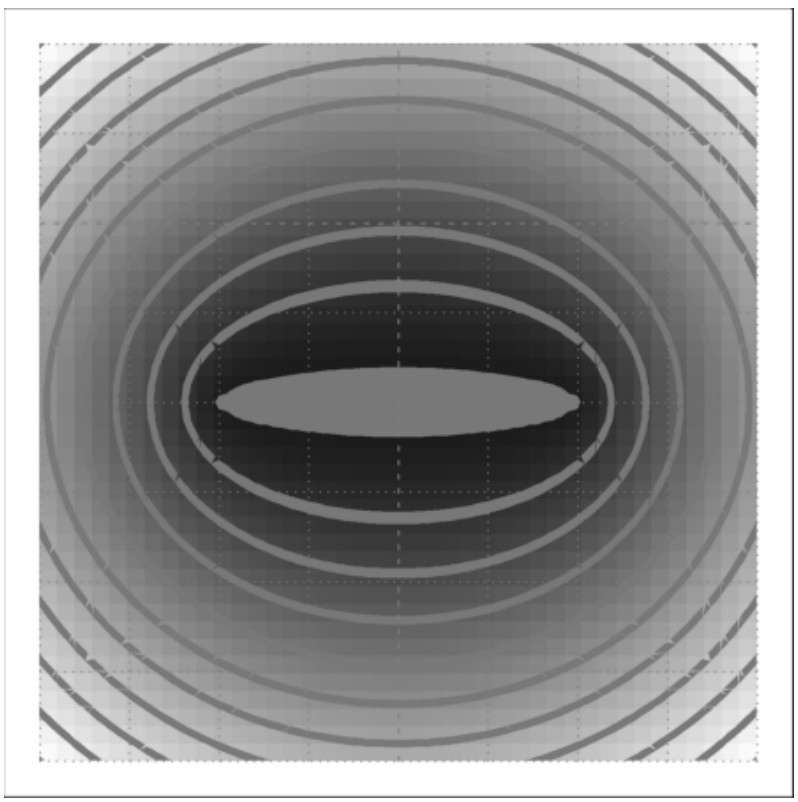


Tentu saja, nilai minimum 0 dicapai di A.

Dua titik

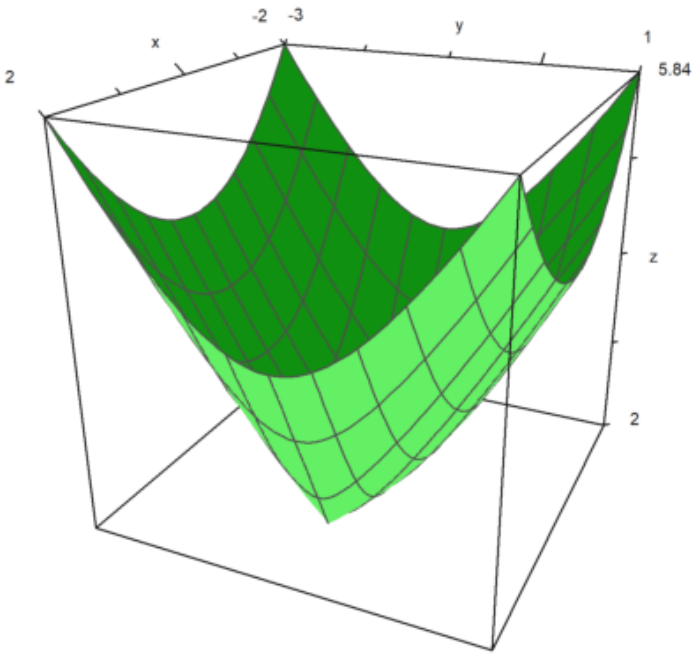
Sekarang kita melihat fungsi MA+MB di mana A dan B adalah dua titik (tetap). Merupakan "fakta yang sudah dikenal" bahwa kurva levelnya adalah elips, dengan titik fokus A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

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



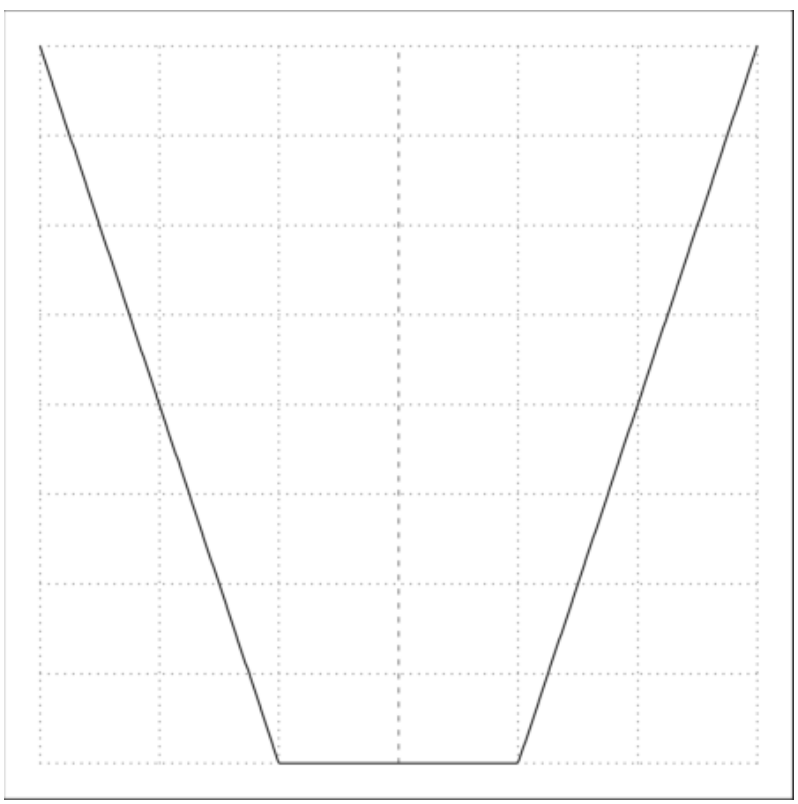
Grafiknya lebih menarik:

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



Pembatasan pada garis (AB) lebih terkenal:

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

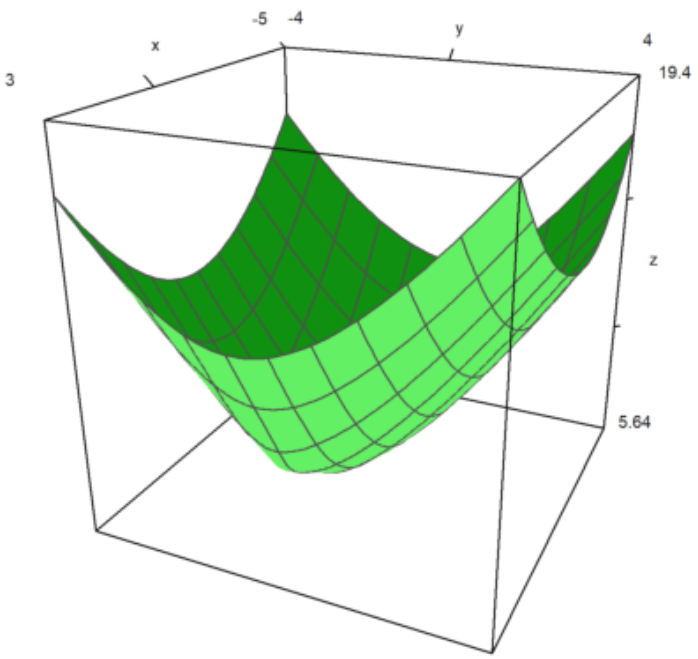


Sekarang, situasinya tidak sesederhana sebelumnya: Sedikit kurang dikenal bahwa MA+MB+MC mencapai nilai minimumnya di satu titik di bidang, tetapi menentukan titik ini tidaklah mudah:

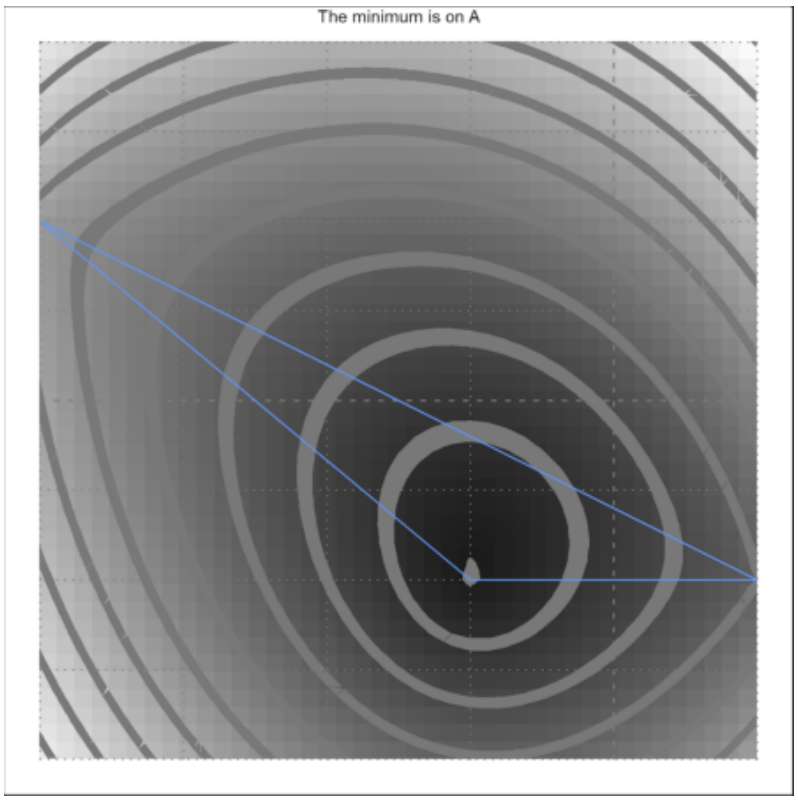
1) Jika salah satu sudut segitiga ABC lebih dari 120° (misalnya di A), maka nilai minimum dicapai tepat di titik ini (misalnya AB+AC).

Contoh:

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

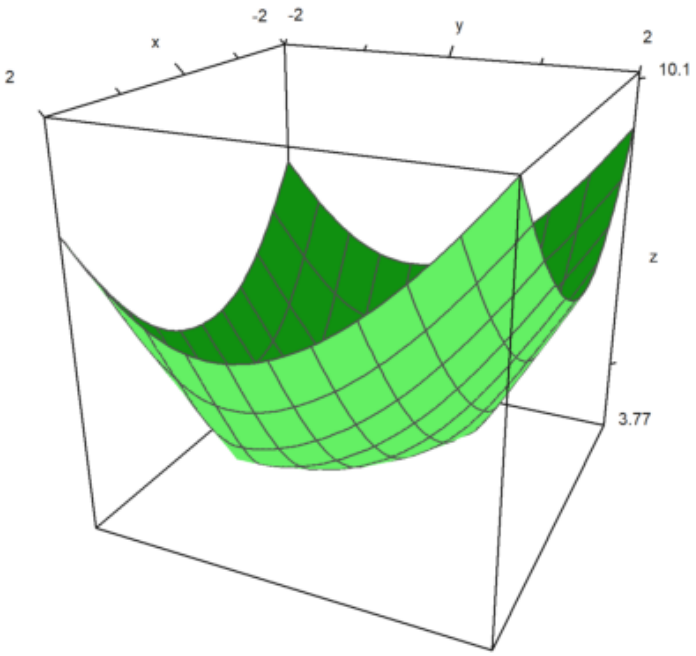



```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

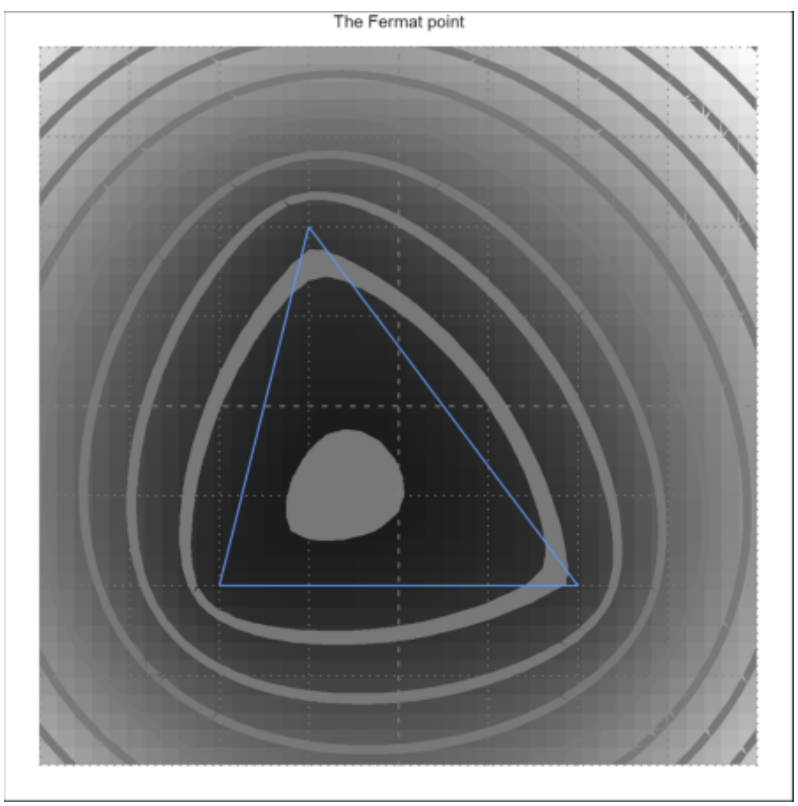


2) Tetapi jika semua sudut segitiga ABC kurang dari 120°, nilai minimum berada pada titik F di dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (masing-masing 120°):

```
>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;
```



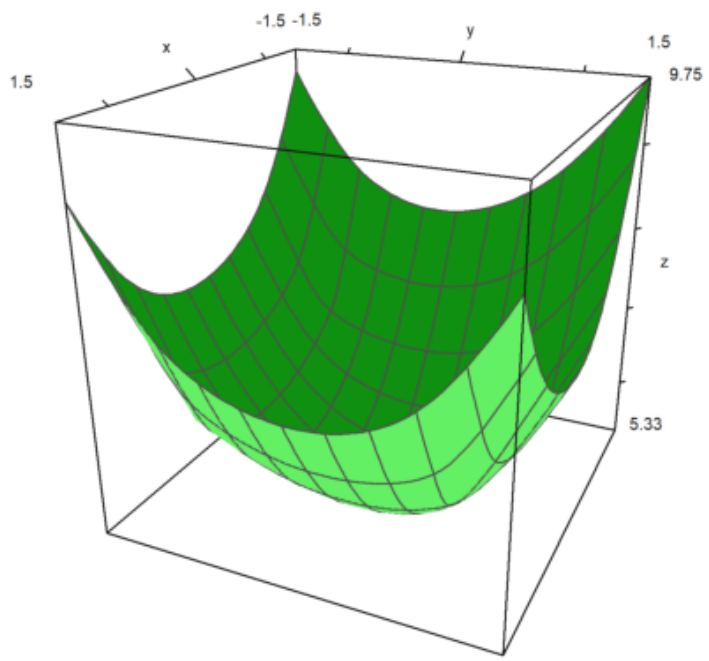
Ini adalah kegiatan yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu sebuah perangkat lunak yang ditulis dalam Java yang memiliki perintah "garis kontur"...

Semua hal di atas telah ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; ia menulis surat kepada para penggemar lain seperti imam Marin Mersenne dan Blaise Pascal yang bekerja di pajak penghasilan. Jadi titik unik F 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...

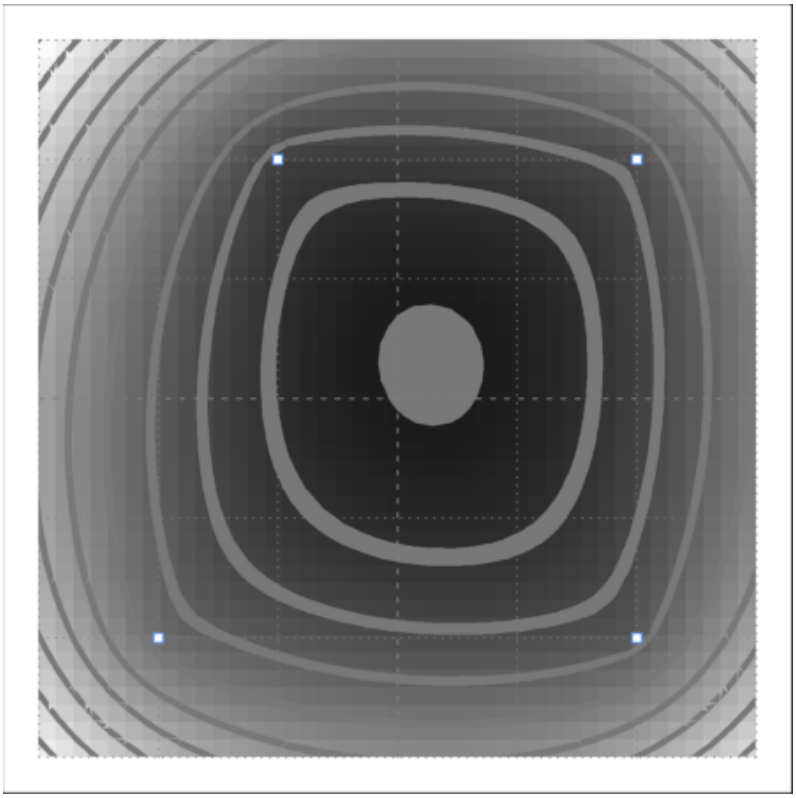
Empat titik

Langkah selanjutnya adalah menambahkan titik ke-4 D dan mencoba meminimalkan MA+MB+MC+MD; katakanlah Anda seorang operator TV kabel dan ingin mengetahui di bidang mana Anda harus menempatkan antena sehingga dapat melayani empat desa dengan 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;
```



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 "garis kontur"...

Semua hal di atas telah ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; ia menulis surat kepada para penggemar lain seperti imam Marin Mersenne dan Blaise Pascal yang bekerja di pajak penghasilan. Jadi titik unik F 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...

Empat titik

Langkah selanjutnya adalah menambahkan titik ke-4 D dan mencoba meminimalkan MA+MB+MC+MD; katakanlah Anda seorang operator TV kabel dan ingin mengetahui di bidang mana Anda harus menempatkan antena sehingga dapat melayani empat desa dengan panjang kabel sesedikit mungkin!

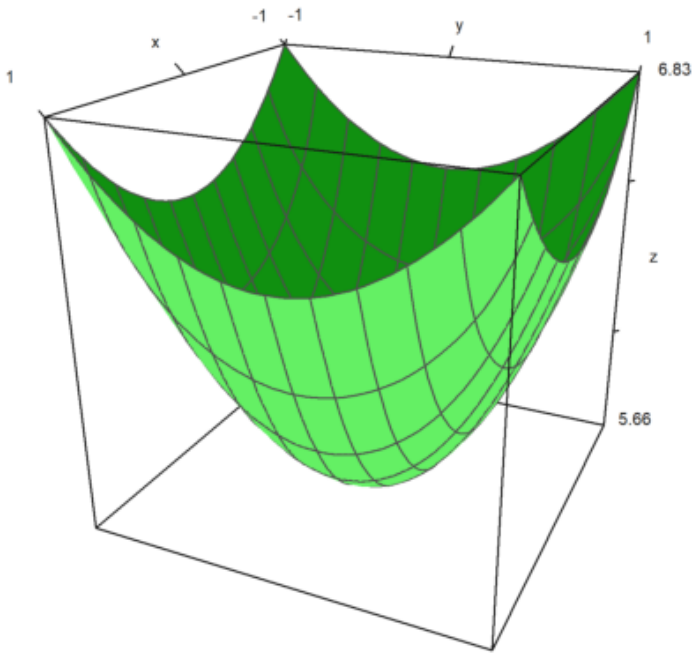
```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```

```
  [0.142858,  0.142857]
```

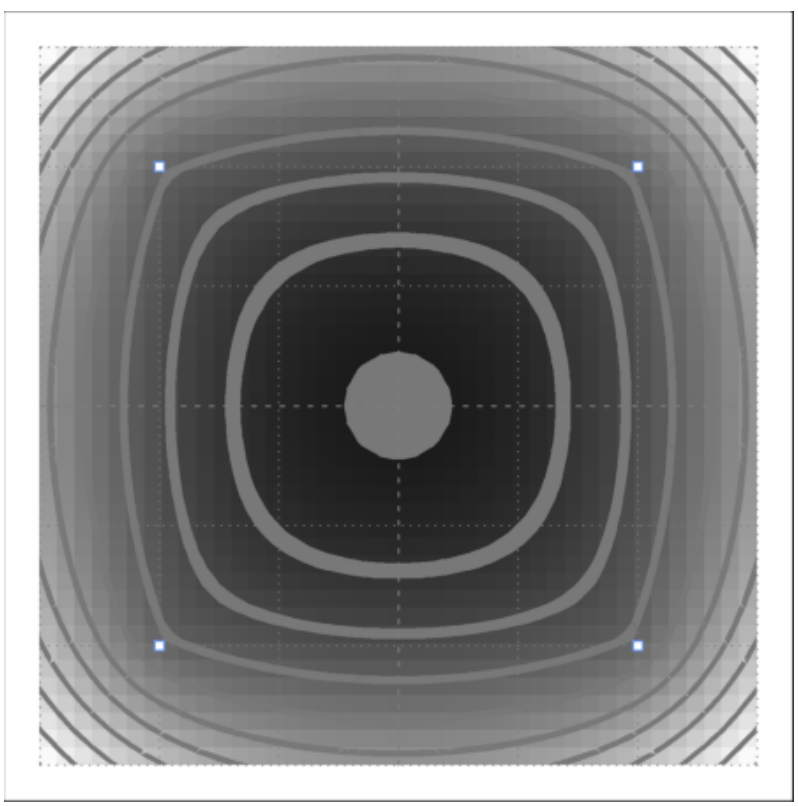
Tampaknya dalam kasus ini, koordinat titik optimal bersifat rasional atau mendekati rasional...

Sekarang ABCD adalah persegi, kita mengharapkan titik optimal 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 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, dua garis yang membentuk kerucut.

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

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

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

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

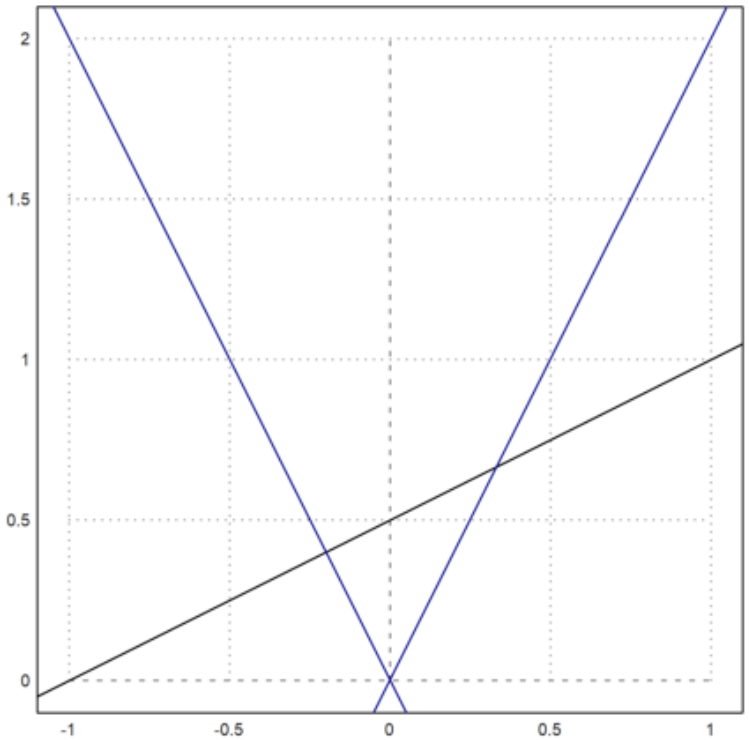
Kemudian tambahkan garis ketiga.

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

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

Kita sudah memplot semuanya sejauh ini.

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



Sekarang kita ambil sebuah titik umum pada sumbu y.

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

$$[0,\ u]$$

Hitung jarak ke g1.

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

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

Hitung jaraknya 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 jarak-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 temukan kedua pusat serta kedua jaraknya.

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

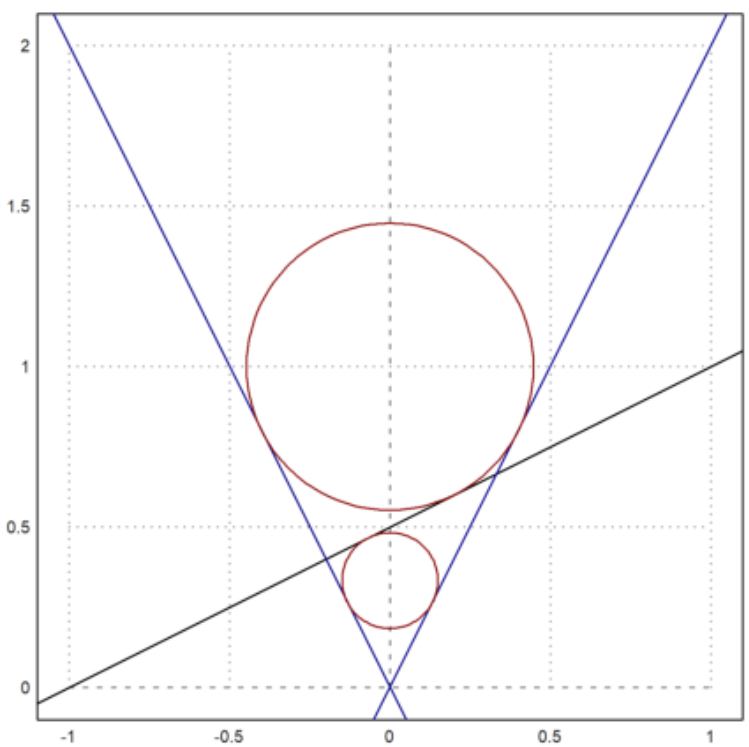
```
  [0.333333,  1]
```

```
>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. Perlu dicatat bahwa Anda dapat mengubah perintah apa pun dalam urutan perintah Povray berikut, dan jalankan 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 adegannya dengan tepat.

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

Selanjutnya kita tulis dua bola ke file Povray.

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

```
  u is not a variable!
  Error in:
  writeln(povsphere([0,0,u[1]],dd[1],povlook(red)); ...
                              ^
```

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

```
  u is not a variable!
  Error in:
  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)));
```

```
  Variable a not found!
  Error in:
  writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)); ...
                                    ^
```

Kita buat bidang yang dibatasi oleh kerucut.

```
>gp=g();
```

```
  Function g not found.
  Try list ... to find functions!
  Error in:
  gp=g(); ...
        ^
```

```
>pc=povcone([0,0,0],0,[0,0,a],1,"");
```

```
  Variable a not found!
  Error in:
  pc=povcone([0,0,0],0,[0,0,a],1,""); ...
                              ^
```

```
>vp=[gp[1],0,gp[2]]; dp=gp[3];
```

```
  gp is not a variable!
  Error in:
  vp=[gp[1],0,gp[2]]; dp=gp[3]; ...
           ^
```

```
>writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```

```
  Variable or function vp not found.
  Error in:
  writeln(povplane(vp,dp,povlook(blue,0.5),pc)); ...
                   ^
```

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

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

```
  u is not a variable!
  Error in:
  P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]); ...
                           ^
```

```
>writeln(povpoint(P1,povlook(yellow)));
```

```
  Variable or function P1 not found.
  Error in:
  writeln(povpoint(P1,povlook(yellow))); ...
                   ^
```

```
>P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
```

```
  u is not a variable!
  Error in:
  P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]); ...
                           ^
```

```
>writeln(povpoint(P2,povlook(yellow)));
```

```
  Variable or function P2 not found.
  Error in:
  writeln(povpoint(P2,povlook(yellow))); ...
                      ^
```

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

```
>P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
```

```
  u is not a variable!
  Error in:
  P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]]; ...
                             ^
```

```
>writeln(povpoint(P3,povlook(yellow)));
```

```
  Variable or function P3 not found.
  Error in:
  writeln(povpoint(P3,povlook(yellow))); ...
                      ^
```

```
>P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
```

```
  u is not a variable!
  Error in:
  P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]]; ...
                             ^
```

```
>writeln(povpoint(P4,povlook(yellow)));
```

```
  Variable or function P4 not found.
  Error in:
  writeln(povpoint(P4,povlook(yellow))); ...
                      ^
```

Selanjutnya kita hitung perpotongan P1P2 dengan bidang.

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

```
  Variable or function vp not found.
  Error in:
  t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P ...
             ^
```

```
>writeln(povpoint(P5,povlook(yellow)));
```

```
  Variable or function P5 not found.
  Error in:
  writeln(povpoint(P5,povlook(yellow))); ...
                      ^
```

Kita hubungkan titik-titik tersebut dengan segmen garis.

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

```
  Variable or function P1 not found.
  Error in:
  writeln(povsegment(P1,P2,povlook(yellow))); ...
                         ^
```

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

```
  Variable or function P5 not found.
  Error in:
  writeln(povsegment(P5,P3,povlook(yellow))); ...
                         ^
```

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

```
  Variable or function P5 not found.
  Error in:
  writeln(povsegment(P5,P4,povlook(yellow))); ...
                         ^
```

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

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);
```

```
  Variable a not found!
  Error in:
  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);
```

```
  P1 is not a variable!
  Error in:
  pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defa ...
                               ^
```

```
>writeln(povintersection([pcw,pc1],povlook(gray)));
```

```
  Variable pcw not found!
  Error in:
  writeln(povintersection([pcw,pc1],povlook(gray))); ...
                               ^
```

```
>pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
```

```
  P2 is not a variable!
  Error in:
  pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defa ...
                              ^
```

```
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

```
  Variable pcw not found!
  Error in:
  writeln(povintersection([pcw,pc2],povlook(gray))); ...
                               ^
```

Jalankan program Povray.

```
>povend();
```

Untuk mendapatkan Anaglyph dari ini, kita perlu memasukkan semuanya ke dalam fungsi scene. 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
```

Kamu memerlukan kacamata merah/cyan untuk menikmati efeknya.

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

```
  Global variable a not found in "global" command.
  scene:
      global a,u,dd,g,g1,defaultpointsize;
  Try "trace errors" to inspect local variables after errors.
  povanaglyph:
      scene$(args());
```

Contoh 8: Geometri Bumi

Di notebook ini, kita ingin melakukan beberapa perhitungan bola. Fungsi-fungsinya terdapat di file "spherical.e" dalam folder contoh. Kita perlu memuat file itu terlebih dahulu.

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

Untuk memasukkan posisi geografis, kita menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut 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 (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 hitung vektor dari satu kota ke kota lain pada bola ideal. Vektor ini adalah [heading, jarak] dalam radian. Untuk menghitung jarak di 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-So
```

```
  65°20’26.60’’
  53.8945384608
```

Ini merupakan pendekatan yang baik. Rutinitas berikut menggunakan pendekatan yang lebih akurat. Pada jarak yang pendek seperti ini, hasilnya hampir sama.

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

```
  Commands must be separated by semicolon or comma!
  Found:   (character 32)
  You can disable this in the Options menu.
  Error in:
  esdist(FMIPA,Semarang)->" km"  ...
                              ^
```

Terdapat sebuah fungsi untuk arah (heading), dengan memperhitungkan bentuk elips bumi. Sekali lagi, kita mencetaknya dengan cara yang lebih canggih.

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

```
       65.34°
```

Sudut sebuah segitiga bisa melebihi 180° pada permukaan bola.

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

```
  180°0’10.77’’
```

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

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

```
  2116.02948749 km^2
```

Ada sebuah fungsi untuk ini, yang menggunakan rata-rata lintang 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 bisa menambahkan vektor ke posisi. Sebuah vektor berisi heading dan jarak, keduanya dalam radian. Untuk membuat vektor, kita menggunakan svector. Untuk menambahkan vektor ke posisi, kita menggunakan saddvector.

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

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

Fungsi-fungsi ini mengasumsikan bola ideal. Sama dengan bumi.

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

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

Mari kita beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (dengan 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 cukup akurat.

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

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

Arah (heading) sama dengan yang dihitung di Google Earth.

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

```
  294°17’2.85’’
```

Namun, kita tidak lagi mendapatkan posisi tujuan yang tepat jika menambahkan arah (heading) dan jarak ke posisi awal. Hal ini terjadi karena kita tidak menghitung fungsi inversnya 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’
```

Erornya tidak besar, hanya saja

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

```
  S 6°10.500’ E 106°48.717’
```

Tentu saja, kita tidak bisa berlayar dengan arah yang sama dari satu tujuan ke tujuan lain jika ingin menempuh jalur terpendek. Bayangkan, Anda terbang ke arah timur laut (NE) mulai dari sembarang titik di bumi. Maka Anda akan melingkar menuju kutub utara. Lingkaran besar (great circles) tidak mengikuti arah tetap!

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

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

Sekarang kita menambahkan 10 kali sepersepuluh jarak, menggunakan arah menuju Monas, yang kita dapatkan di Tugu.

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

Hasilnya jauh meleset.

```
>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 dengan lintang yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

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

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

```
       79.69°
```

Namun, jika kita mengikuti arah kompas ini, kita akan memutar spiral menuju kutub utara! Jadi kita harus menyesuaikan arah sepanjang perjalanan. Untuk perkiraan kasar, kita menyesuaikannya setiap 1/10 dari jarak total.

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

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

Jarak-jarak tidak tepat, karena kita akan menambahkan sedikit kesalahan jika mengikuti arah yang sama terlalu lama.

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

```
       0.203km
```

Kita mendapatkan perkiraan yang cukup baik, jika kita menyesuaikan arah 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 deretan posisi GPS sepanjang lingkaran besar menuju Monas menggunakan 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’
```

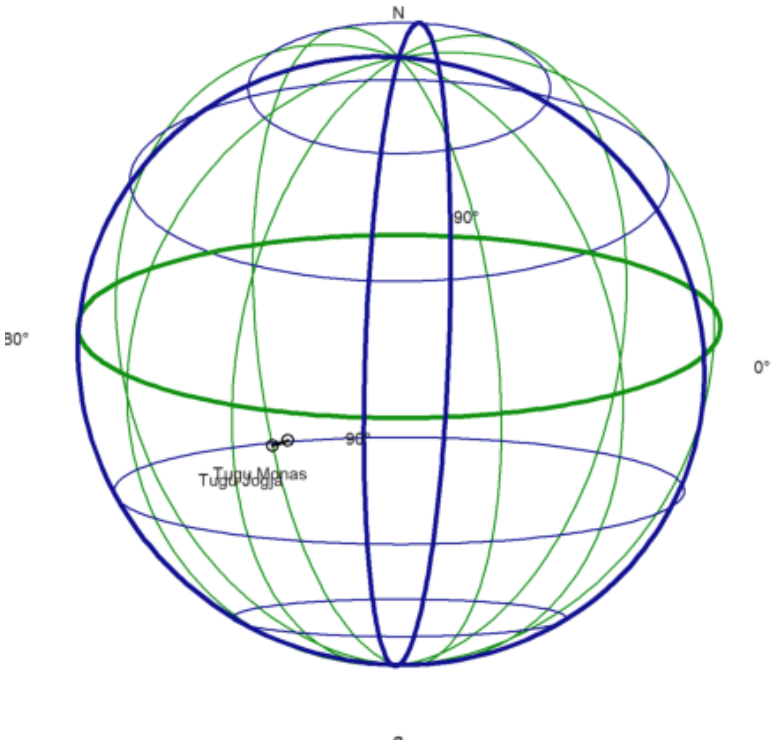
Kita menulis sebuah fungsi, yang menggambar bumi, kedua posisi, serta posisi-posisi di antaranya.

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

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

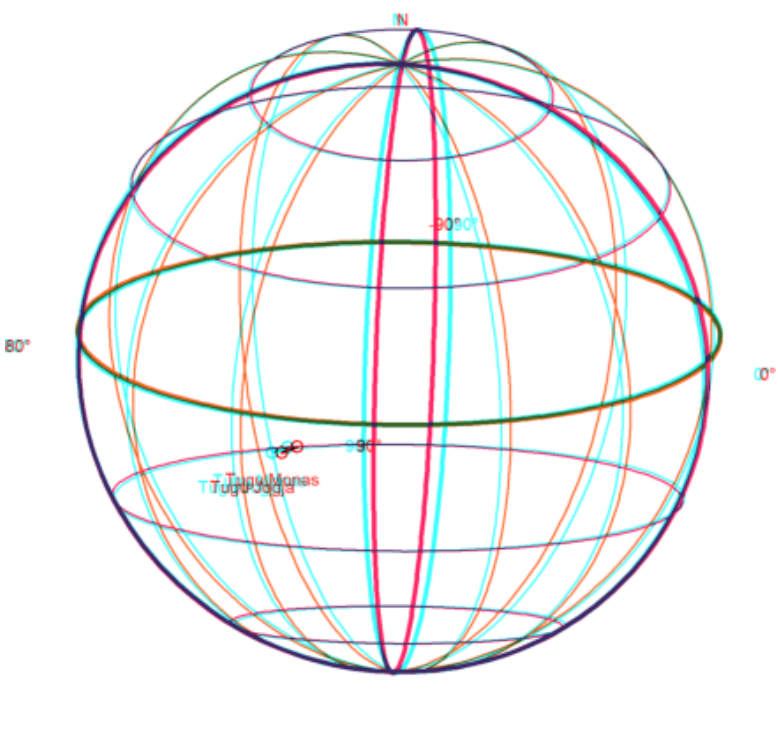
Sekarang gambar semuanya.

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



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

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



Latihan

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

Petunjuk:

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

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

Petunjuk:

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

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

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

merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).

- Suatu segi-4 merupakan segi-4 garis singgung apabila keempat garis bagi sudutnya

bertemu di satu titik.

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

panjang sisi-sisi yang berhadapan sama.

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

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